UNIVERSITY OF THE BASQUE COUNTRY
(UPV/EHU)

DOCTORAL THESIS

---

# Pseudospectral Methods for the Fractional Laplacian on $\mathbb{R}$

---

*Author:*
Jorge Enrique Cayama Mendoza

*Advisors:*
Prof. Carlota María Cuesta Romero
Prof. Francisco de la Hoz Méndez

*A thesis submitted in fulfillment of the requirements for the degree of Doctor of Philosophy*

*in the*

Department of Mathematics

May 25, 2020


Universidad del País Vasco
Euskal Herriko Unibertsitatea

# Declaration

I hereby declare that except where specific reference is made to the work of others, the contents of this thesis are original and have not been submitted in whole or in part for consideration for any other degree or qualification in this, or any other university.

Jorge Enrique Cayama Mendoza

May, 2020

UNIVERSITY OF THE BASQUE COUNTRY (UPV/EHU)

# *Abstract*

Faculty of Science and Technology

Department of Mathematics

Doctor of Philosophy

**Pseudospectral Methods for the Fractional Laplacian on $\mathbb{R}$**

by Jorge Enrique Cayama Mendoza

In this thesis, first, we propose a novel pseudospectral method to approximate accurately and efficiently the fractional Laplacian without using truncation. More precisely, given a bounded regular function defined over $\mathbb{R}$, we map the unbounded domain into a finite one, and represent the resulting function as a trigonometric series. Therefore, a key ingredient is the computation of the fractional Laplacian of an elementary trigonometric function. As an application of the method, we do the simulation of Fisher's equation with the fractional Laplacian in the monostable case.

In addition, using complex variable techniques, we compute explicitly, in terms of the ${}_2F_1$ Gaussian hypergeometric function, the one-dimensional fractional Laplacian of the Higgins functions, the Christov functions, and their sine-like and cosine-like versions. After discussing the numerical difficulties in the implementation of the proposed formulas, we develop another method that gives exact results, by using variable precision arithmetic.

Finally, we discuss some other numerical approximations of the fractional Laplacian using a fast convolution technique. While the resulting techniques are less accurate, they are extremely fast; furthermore, the results can be improved by the use of Richardson's extrapolation.

# *Acknowledgements*

Undertaking this Ph.D. program has been a truly life-changing experience for me and the work presented in this thesis would not have been possible without my close associations with many individuals. I take this opportunity to extend my gratitude and appreciation to all those who made this work become a reality.

Above all, I would like to thank God for giving me good health and the strength to finish this study.

I wish to express my sincere appreciation to my advisors, Prof. Carlota María Cuesta and Prof. Francisco de la Hoz. They convincingly guided and encouraged me to be professional and do the right thing even when the road got tough. Without their persistent help, patience, encouragement, and immense knowledge, the goal of this project would not have been realized.

I would also like to thank Prof. Carlos García Cervera for his valuable and constructive suggestions as director of my three-month stay at the UCSB, and also as reviewer of this manuscript. His willingness to give his time so generously has been very much appreciated. My grateful thanks are also extended to Prof. Héctor Ceniceros and Prof. Gustavo Ponce for their assistance and guidance while I was working at the UCSB.

My sincere thanks also go to Prof. Luis Vega, who provided me an opportunity to join the Linear and Non-linear Waves research line as a Ph.D. Student at BCAM. Without his precious support, it would not have been possible to conduct the last stage of this research.

I am grateful to Prof. Benito Chen-Charpentier and Prof. David Lannes, who also appear as reviewers of this work.

This dissertation would not have materialized without the financial support of the Spanish Government through the grant BES-2015-071231, and without the ERCEA Advanced Grant 2014 669689-HADE, for the last stage of this research.

ERCEA Advanced Grant 2014 669689-HADE.

My thanks and appreciations also go to my colleagues and people at the UPV/EHU who have willingly helped me out with their companionship.

Last but not least, I would like to express my deepest gratitude to my family and friends for their warm love, continued patience, and endless support.

# Contents

# List of Figures

# List of Tables

*Dedicated to my dad…*

# Resumen en castellano

En esta tesis doctoral se propone un nuevo método pseudoespectral para aproximar de manera precisa y eficaz el laplaciano fraccionario $(-\Delta)^{\alpha/2}$ de una función regular $u(x)$ definida sobre la recta real. Este operador es definido para $\alpha \in (0,\, 2)$ como (ver [1]):

$$(-\Delta)^{\alpha/2} u(x) = c_\alpha \int_{\mathbb{R}} \frac{u(x) - u(y)}{|x-y|^{1+\alpha}}\, dy, \tag{1}$$

con

$$c_\alpha = \frac{2^\alpha \Gamma(\frac{1+\alpha}{2})}{\sqrt{\pi}|\Gamma(-\frac{\alpha}{2})|}. \tag{2}$$

La representación en el espacio de Fourier es

$$\widehat{(-\Delta)^{\alpha/2} u}(\xi) = |\xi|^\alpha \hat{u}(\xi). \tag{3}$$

En vez de trabajar directamente con la definición (1), se hace uso de la siguiente representación, la cual viene dada como un lema y, además, requiere de acotamiento y regularidad $\mathcal{C}^2$:

$$(-\Delta)^{\alpha/2} u(x) = \begin{cases} \dfrac{1}{\pi} \displaystyle\int_{-\infty}^{\infty} \frac{u_x(y)}{x-y} dy, & \alpha = 1, \\ \dfrac{c_\alpha}{\alpha(1-\alpha)} \displaystyle\int_{-\infty}^{\infty} \frac{u_{xx}(y)}{|x-y|^{\alpha-1}} dy, & \alpha \neq 1. \end{cases} \tag{4}$$

Uno de los objetivos principales de esta tesis es contemplar el tratamiento numérico del operador (4) sobre todo el dominio $\mathbb{R}$, por lo que se debe tener un cuidado particular en el estudio de dominios no acotados. Es por ello que se opta por implementar un mapeo algebraico que transforma el dominio infinito en uno finito. Más aún, si se considera $x \in \mathbb{R}$, $\psi \in [-1,\, 1]$, y $s \in [0,\, \pi]$, entonces los dominios se transforman de unos a otros con las siguientes relaciones [2]:

$$\psi = \frac{x}{\sqrt{L^2 + x^2}} \Longleftrightarrow x = \frac{L\psi}{\sqrt{1-\psi^2}}$$

y

$$x = L\cot(s) \Longleftrightarrow s = \operatorname{arccot}\left(\frac{x}{L}\right),$$

donde $L$ es un factor de escala. La $n$-ésima función racional de Chebyshev es:

$$TB_n(x) \equiv T_n(\psi) \equiv \cos(nt), \quad \forall n \in \mathbb{N} \cup \{0\}.$$

De esta forma,

$$TB_n(x) = T_n\left(\frac{x}{\sqrt{L^2+x^2}}\right) = \cos\left(n\,\mathrm{arccot}\left(\frac{x}{L}\right)\right), \quad \forall n \in \mathbb{N}\cup\{0\}, \tag{5}$$

donde $T_n(\psi)$ es el $n$-ésimo polinomio de Chebyshev. Entonces, (5) es reescrito como

$$T_n(\psi) = TB_n(x) = \cos(ns).$$

Por lo tanto, una expansión en términos de los polinomios de Chebyshev o de las funciones racionales de Chebyshev es equivalente a una expansión en serie de cosenos. Es preferible trabajar con una representación trigonométrica que permita un mejor manejo de estos polinomios, por lo que se considera el siguiente cambio de variable:

$$\psi = \cos(s) \Longleftrightarrow x = L\cot(s), \quad s \in [0,\pi]. \tag{6}$$

Luego, (4) se transforma a través de la relación $x = L\cot(s)$, con lo cual se evita truncar el dominio en su evaluación numérica. Entonces, una forma equivalente de expresar (4) en la nueva variable espacial $s \in [0,\ \pi]$ es:

$$(-\Delta)^{\alpha/2}u(s) = \begin{cases} \dfrac{\sin(s)}{L\pi}\displaystyle\int_0^\pi \frac{\sin(\eta)u_s(\eta)}{\sin(s-\eta)}d\eta, & \alpha = 1, \\ \dfrac{c_\alpha|\sin(s)|^{\alpha-1}}{L^\alpha\alpha(1-\alpha)} & \\ \cdot\displaystyle\int_0^\pi \frac{\sin^\alpha(\eta)(\sin(\eta)u_{ss}(\eta)+2\cos(\eta)u_s(\eta))}{|\sin(s-\eta)|^{\alpha-1}}d\eta, & \alpha \neq 1. \end{cases} \tag{7}$$

Se define la siguiente familia de nodos que discretizará el dominio $[0,\ \pi]$:

$$s_j = \frac{\pi(2j+1)}{2N},$$

donde $0 \leq j \leq N-1$, lo cual divide a dicho intervalo en $N$ partes igualmente espaciadas. Al escribir $u(s_j)$ se hace referencia a $u(x(s))$ evaluada en $s_j$, y, con un poco de abuso de notación, $u(s_j) \equiv u(x_j)$. Por otro lado, como ya fue mencionado, $u(s)$ puede representarse como una expansión en serie de cosenos. Sin embargo, se considera una serie más general conformada por $e^{iks}$, con $k \in \mathbb{Z}$, lo cual permite alcanzar una mayor facilidad en la implementación numérica:

$$u(s) = \sum_{k=-\infty}^{\infty} \hat{u}(k)e^{iks}, \quad s \in [0,\pi].$$

Ahora, para la obtención de los coeficientes $\hat{u}(k)$, se debe extender el dominio de $u(s)$ de $s \in [0,\pi]$ a $s \in [0,\ 2\pi]$; para ello se elige una extensión par de $u(s)$ en $s = \pi$, utilizando la familia de $2N$ nodos $s_j = \pi(2j+1)/(2N)$, $0 \leq j \leq 2N-1$. Posteriormente, se trunca la serie considerando sólo $2N$ términos para una función

$u$ de período $2\pi$:

$$u(s) \approx \sum_{k=-N}^{N-1} \hat{u}(k) e^{iks}, \quad s \in [0,\, 2\pi]. \tag{8}$$

Así, se puede adoptar una técnica pseudoespectral para obtener de manera unívoca los $2N$ coeficientes $\hat{u}(k)$ en (8), es decir, para $s_j$ se tiene la siguiente identidad:

$$\begin{aligned} u(s_j) &\equiv \sum_{k=-N}^{N-1} \hat{u}(k) e^{iks_j} = \sum_{k=-N}^{N-1} \hat{u}(k) e^{ik\pi(2j+1)/(2N)} \\ &= \sum_{k=0}^{N-1} \left[\hat{u}(k) e^{ik\pi/(2N)}\right] e^{2ijk\pi/(2N)} \\ &\quad + \sum_{k=N}^{2N-1} \left[\hat{u}(k-2N) e^{i(k-2N)\pi/(2N)}\right] e^{2ijk\pi/(2N)}. \end{aligned} \tag{9}$$

Análogamente,

$$\hat{u}(k) \equiv \frac{e^{-ik\pi/(2N)}}{2N} \sum_{j=0}^{2N-1} u(s_j) e^{-2ijk\pi/(2N)}. \tag{10}$$

Las transformadas de Fourier (9) y (10) se pueden calcular de forma muy eficiente a través de la transformada rápida de Fourier [3]. Ahora bien, con la aproximación de $u(s)$ por (8), el problema se reduce al cálculo de $(-\Delta)^{\alpha/2} e^{iks}$, lo cual viene dado como un teorema y presenta la siguiente expresión explícita para $\alpha \in (0,1) \cup (1,2)$:

$$(-\Delta)^{\alpha/2}(e^{iks}) = \begin{cases} \dfrac{c_\alpha |\sin(s)|^{\alpha-1}}{8L^\alpha \tan(\frac{\pi\alpha}{2})} \displaystyle\sum_{l=-\infty}^{\infty} e^{i2ls}((1-\alpha)k^2 - 4kl) \\ \cdot \dfrac{\Gamma\left(\frac{-1+\alpha}{2} + |l|\right)\Gamma\left(\frac{-1-\alpha}{2} + \left|\frac{k}{2} - l\right|\right)}{\Gamma\left(\frac{3-\alpha}{2} + |l|\right)\Gamma\left(\frac{3+\alpha}{2} + \left|\frac{k}{2} - l\right|\right)}, & k \text{ par}, \\ i\dfrac{c_\alpha |\sin(s)|^{\alpha-1}}{8L^\alpha} \displaystyle\sum_{l=-\infty}^{\infty} e^{i2ls}((1-\alpha)k^2 - 4kl) \\ \cdot \operatorname{sgn}(\frac{k}{2} - l) \dfrac{\Gamma\left(\frac{-1+\alpha}{2} + |l|\right)\Gamma\left(\frac{-1-\alpha}{2} + \left|\frac{k}{2} - l\right|\right)}{\Gamma\left(\frac{3-\alpha}{2} + |l|\right)\Gamma\left(\frac{3+\alpha}{2} + \left|\frac{k}{2} - l\right|\right)}, & k \text{ impar}. \end{cases} \tag{11}$$

Asimismo, cuando $\alpha = 1$,

$$(-\Delta)^{1/2}(e^{iks}) = \begin{cases} \dfrac{|k| \sin^2(s)}{L} e^{iks}, & k \text{ par}, \\ \dfrac{ik}{L\pi}\left(\dfrac{-2}{k^2-4} - \displaystyle\sum_{l=-\infty}^{\infty} \frac{4\operatorname{sgn}(l) e^{i2ls}}{(k-2l)((k-2l)^2-4)}\right), & k \text{ impar}. \end{cases} \tag{12}$$

Desde un punto de vista computacional, es conveniente construir una matriz operacional $\mathbf{M}_\alpha \in \mathcal{M}_{(2N)\times(2N)}(\mathbb{C})$ basada en las identidades (11) y (12):

$$\begin{pmatrix} (-\Delta)^{\alpha/2}u(s_0) \\ \vdots \\ (-\Delta)^{\alpha/2}u(s_{2N-1}) \end{pmatrix} \approx \mathbf{M}_\alpha \cdot \begin{pmatrix} \hat{u}(0) \\ \vdots \\ \hat{u}(N-1) \\ \hat{u}(-N) \\ \vdots \\ \hat{u}(-1) \end{pmatrix}. \tag{13}$$

Cabe destacar que, para $k \in \{-N, \dots N-1\}$, esta matriz se crea y se almacena una sola vez, puede ser usada cuantas veces se requiera y su construcción se simplifica utilizando algunas propiedades de simetría, reduciéndose a considerar sólo $k \in \{1, \dots, N-1\}$.

Luego de discutir sobre la convergencia del método, se presenta un experimento numérico para medir la exactitud del esquema desarrollado. Para ello se consideran, en un principio, las siguientes dos funciones con decaimiento polinomial:

$$u_1(x) = \frac{x^2-1}{x^2+1}, \qquad u_2(x) = \frac{2x}{x^2+1},$$

donde la primera tiende a 1 como $\mathcal{O}(1/x^2)$, mientras que la segunda tiende a 0 como $\mathcal{O}(1/x)$. Un cambio de variable como $x = \cot(s)$, con $L = 1$, las convierte respectivamente en $u_1(s) = \cos(2s)$ y $u_2(s) = \sin(2s)$, esto es, la parte real e imaginaria de $e^{i2s}$, por lo que se va a calcular el laplaciano fraccionario de $u_1(x) + iu_2(x)$, cuyo resultado analítico viene dado como

$$(-\Delta)^{\alpha/2}(u_1(x) + iu_2(x)) = -\frac{2\Gamma(1+\alpha)}{(1+ix)^{1+\alpha}},$$

o, en la variable $s$,

$$(-\Delta)^{\alpha/2}(e^{i2s}) = -\frac{2\Gamma(1+\alpha)}{(1+i\cot(s))^{1+\alpha}} = -2\Gamma(1+\alpha)(-i\sin(s)e^{is})^{1+\alpha}. \tag{14}$$

La exactitud del método se obtiene al comparar este resultado analítico con el resultado numérico, obteniéndose el error a través de la norma $L^\infty$ como una función de $\alpha$. Se muestran los errores máximos globales y los parámetros necesarios para alcanzar un error del orden de $10^{-13}$. Adicionalmente, se toma como un segundo ejemplo una función con decaimiento gaussiano,

$$u_3 = \exp(-x^2),$$

tal que

$$(-\Delta)^{\alpha/2}u_3(x) = \frac{2^\alpha\Gamma(1/2+\alpha/2)}{\sqrt{\pi}}{}_1F_1(1/2+\alpha/2,\ 1/2,\ -x^2), \tag{15}$$

donde ${}_1F_1$ es la función hipergeométrica confluente (ver [4, Cap. 13]). Acá también se muestran los parámetros para alcanzar un error del orden de $10^{-13}$, ya sea para una extensión par o impar del dominio de definición de la función $u$. Como una aplicación del método, se presenta la simulación de la ecuación de Fisher con el operador laplaciano fraccionario:

$$\partial_t u + (-\Delta)^{\alpha/2} u = f(u), \quad x \in \mathbb{R}, \ t \geq 0, \tag{16}$$

contemplando el caso monoestable, es decir, con el siguiente término fuente no lineal:

$$f(u) = u(1-u). \tag{17}$$

El interés en esta simulación yace en el hecho de que (16) exhibe frentes de aceleración y el esquema acá presentado permite una captura de esta propagación rápida. La estabilidad dinámica en la solución de estas ondas viajeras ha sido estudiada por numerosos autores (ver por ejemplo [5]). En esta tesis, la aplicación se limita a usar una condición inicial de decaimiento lento. En todos los experimentos numéricos, se considera una extensión par en $s = \pi$, y se utiliza el método clásico de Runge-Kutta de cuarto orden para el avance en el tiempo (ver [6, p. 226]). Las simulaciones se han hecho para distintos valores de $\alpha$, y se comprueba numéricamente que para cada valor de $\alpha$ se tiene un frente de onda cuya velocidad sigue un patrón exponencial con respecto al tiempo y, además, cada curva integral presenta un decaimiento potencial en sus colas. Se deja de manera introductoria lo que puede significar un desarrollo numérico del laplaciano fraccionario en $n$ dimensiones.

Por otro lado, se realiza un estudio más teórico en donde, mediante un cálculo analítico que involucra técnicas de variable compleja, se obtienen expresiones del laplaciano fraccionario de funciones de Fourier pares del tipo $e^{i2ks}$, todas ellas en términos de la función hipergeométrica gaussiana ${}_2F_1$. Así, es posible obtener el laplaciano fraccionario de las funciones complejas de Higgins y Christov, respectivamente dadas por [7]:

$$\lambda_n(x) = \left(\frac{ix-1}{ix+1}\right)^n, \quad \text{y} \quad \mu_n(x) = \frac{(ix-1)^n}{(ix+1)^{n+1}}, \quad n \in \mathbb{Z},$$

así como de las siguientes familias de funciones:

- Función Higgins en su forma cosinusoidal:

$$\mathrm{CH}_{2n}(x) = \frac{\lambda_n(x) + \lambda_{-n}(x)}{2}, \quad n = 0, 1, 2, \ldots$$

- Función Higgins en su forma sinusoidal:

$$\mathrm{SH}_{2n+1}(x) = \frac{\lambda_{n+1}(x) - \lambda_{-n-1}(x)}{2i}, \quad n = 0, 1, 2, \ldots$$

- Función Christov en su forma cosinusoidal:

$$\mathrm{CC}_{2n}(x) = \frac{\mu_n(x) - \mu_{-n-1}(x)}{2}, \quad n = 0, 1, 2, \ldots$$

- Función Christov en su forma sinusoidal:

$$\mathrm{SC}_{2n+1}(x) = -\frac{\mu_n(x) + \mu_{-n-1}(x)}{2i}, \quad n = 0, 1, 2, \ldots$$

Luego de realizar una discusión sobre las dificultades y limitaciones numéricas de MATLAB [8] en la implementación de las fórmulas propuestas que contienen a ${}_2F_1$, se desarrolla un método innovador y alternativo a través del uso de aritmética de precisión variable, con el que se tienen resultados exactos y supera ampliamente la implementación de MATLAB. Posteriormente, se testea la exactitud del método desde un punto de vista numérico, comparando los resultados con aquellos obtenidos mediante la matriz operacional (13).

Por último, tomando ideas de [9], se plantean otras aproximaciones numéricas del laplaciano fraccionario en $\mathbb{R}$ con el uso de la técnica de convolución rápida. Más aún, se aplica nuevamente el mapeo algebraico $x = L\cot(s)$ para obtener la integral singular sobre $s \in [0, \pi]$, dada por (7), y se prueba con distintas fórmulas de cuadratura, expresándolas como convoluciones discretas. Además, se hace uso del teorema de convolución discreta, el cual establece que la transformada discreta de Fourier de una convolución discreta es igual al producto de las transformadas discretas de Fourier, esto es, $(\widehat{u * v})_p = \widehat{u}_p \widehat{v}_p$, donde $u = \{u_r\}$ y $v = \{v_r\}$ son dos secuencias de números. Debido al uso de la transformada rápida de Fourier para el cálculo de las transformadas discretas [3], esta técnica se suele denominar convolución rápida, ya que se requieren $\mathcal{O}(N \log N)$ operaciones cuando las dos secuencias $u$ y $v$ son $N$-periódicas. Esto representa una reducción muy notable en el número de operaciones de una suma directa, la cual consta de $\mathcal{O}(N^2)$ operaciones y, por lo tanto, se vuelve más costosa para valores relativamente pequeños de $N$. Siguiendo los argumentos previos, se desarrolla un método para aproximar el laplaciano fraccionario que parece ser de segundo orden para todo $\alpha \in (0, 1) \cup (1, 2)$, por lo que se hace notar el enfoque sobre la rapidez en la ejecución del esquema, en vez de la exactitud de los resultados. Sin embargo, la precisión se mejora fácilmente mediante el uso de la extrapolación de Richardson. Se toma nuevamente como funciones test aquellas con decaimiento polinomial ($u_1(x) + iu_2(x)$) y aquella con decaimiento gaussiano ($u_3(x)$).

Finalmente, se anexa un apéndice en donde se detallan, de forma breve, las principales propiedades de los polinomios de Chebyshev y de las funciones racionales de Chebyshev.

# Chapter 1

# Introduction

In this thesis, we are mainly concerned with the numerical approximation of the fractional Laplacian. This is a singular integral operator that can be defined in several equivalent ways, see e.g [1], but that we take for now to be:

$$(-\Delta)^{\alpha/2}u(x) = c_{n,\alpha}\int_{\mathbb{R}^n}\frac{u(x)-u(y)}{|x-y|^{n+\alpha}}\,dy, \tag{1.1}$$

where $\alpha \in (0,2)$, and

$$c_{n,\alpha} = \frac{2^\alpha\Gamma(\frac{n+\alpha}{2})}{\pi^{n/2}|\Gamma(-\frac{\alpha}{2})|}. \tag{1.2}$$

The fractional Laplacian operator is a generalization of the integer-order Laplacian $\Delta \equiv \partial^2/\partial x_1^2 + \ldots + \partial^2/\partial x_n^2$, with $n$ being the dimension. It appears in a number of applications (see, for instance, [10, Table 1] and its references).

Before we describe the main contents of the thesis, let us start this introduction by putting the operator (1.1) into a wider context. We recall that the Fourier transform of $u$ is

$$\hat{u}(\xi) = \int_{-\infty}^{\infty} u(x)e^{-ix\xi}dx, \tag{1.3}$$

and that the inverse Fourier transform is

$$u(x) = \frac{1}{2\pi}\int_{-\infty}^{\infty}\hat{u}(\xi)e^{ix\xi}d\xi. \tag{1.4}$$

Therefore, another definition of the fractional Laplacian consistent with (1.1) is given by associating the operator with the Fourier symbol:

$$\widehat{(-\Delta)^{\alpha/2}u}(\xi) = |\xi|^\alpha\hat{u}(\xi), \tag{1.5}$$

and, hence, when $\alpha = 2$, we recover $(-\Delta)^{2/2}u = -\Delta u$, whereas, when $\alpha = 0$, $(-\Delta)^0 u = u$. We remark that, in [1], the author indeed considers $-(-\Delta)^{\alpha/2}u$ in the definition of the fractional Laplacian, to make it agree with the integer-order Laplacian, when $\alpha = 2$. The same sign convention is chosen in [11], where an excellent and up-to-date introduction to the topic can be found.

We recall that in one dimension, the Hilbert transform [12] is given by

$$\mathcal{H}(u)(x) = \frac{1}{\pi}\int_{-\infty}^{\infty}\frac{u(y)}{x-y}dy,$$

and that, if $u \in L^2(\mathbb{R})$, then $\widehat{\mathcal{H}(u)}(\xi) = -i\,\mathrm{sgn}(\xi)\hat{u}(\xi)$ (see e.g. [13], [14]). It then follows that, when $\alpha = 1$,

$$\widehat{(-\Delta)^{1/2}u}(\xi) = |\xi|\hat{u}(\xi) = -i\,\mathrm{sgn}(\xi)(i\xi)\hat{u}(\xi) = \widehat{\mathcal{H}(u_x)}(\xi),$$

or, formally,

$$(-\Delta)^{1/2}u(x) = \mathcal{H}(u_x)(x). \tag{1.6}$$

On the other hand, the Riesz transform is a generalization of the Hilbert transform to higher dimensions, but an interpretation similar to (1.6) is a little different. If $u \in L^p(\mathbb{R}^n)$, for $1 \le p < \infty$, then the Riesz transform $R_j u$ is defined (see e.g. [15]) for each $1 \le j \le n$ by

$$R_j u(x) = c_n\,\mathrm{PV}\int_{\mathbb{R}^n}\frac{x_j - y_j}{|x-y|^{n+1}}u(y)\,dy, \quad 1 \le j \le n,$$

where PV denotes the principal value of the integral and

$$c_n = \frac{\frac{n+1}{2}}{\pi^{\frac{n+1}{2}}}.$$

Formally, by working in the Fourier side, it can be seen that $R_j(-\Delta)^{1/2}u(x) = \partial_j u$ for every $j$. This and the relation (1.6) serve as a motivation to define the fractional Laplacian as the inverse of an integral transform. This idea is related to the Riemann-Liouville integral that allows defining fractional derivatives by its inversion.

Indeed, the so-called Riesz potential was introduced by Marcel Riesz [16] while looking for generalizations of the Riemann–Liouville operator. On $\mathbb{R}^n$, it is defined by the convolution with the kernel $c_{n,-\alpha}|x-y|^{-(n-\alpha)}$, with $0 < \alpha < n$:

$$I_\alpha(u)(x) = u * K_\alpha(x) = c_{n,-\alpha}\int_{\mathbb{R}^n}\frac{u(y)}{|x-y|^{n-\alpha}}\,dy. \tag{1.7}$$

We observe that

$$\widehat{\frac{1}{|x|^{n-\alpha}}}(\xi) = {c_{n,-\alpha}}^{-1}(2\pi)^{-\alpha}|\xi|^{-\alpha}. \tag{1.8}$$

Then, from (1.7) and (1.8), we have formally:

$$\widehat{I_\alpha u}(\xi) = (2\pi|\xi|)^{-\alpha}\hat{u}(\xi),$$

On the other hand,

$$\widehat{(-\Delta)\varphi}(\xi) = |\xi|^2 \hat{u}(\xi),$$

which suggests that

$$\widehat{I_\alpha u}(\xi) = (2\pi|\xi|)^{-\alpha} \widehat{(-\Delta)^{-\alpha/2}\varphi}(\xi).$$

In the search for an integral equation that inverts this relationship, one looks at an operator that satisfies:

$$\widehat{(-\Delta)^{\alpha/2}u}(\xi) = |\xi|^\alpha \hat{u}(\xi),$$

with $0 < \alpha \leq 2$ and

$$|\xi|^\alpha = \left[\sum_{j=1}^{n} \xi_j^2\right]^{\alpha/2}.$$

One such operator is precisely the operator (1.1) (see [17–19] for details), where the fractional power is taken over $(-\Delta)$, in order to obtain a positive operator. This formulation of $(-\Delta)^{\alpha/2}$ is also the negative generator of the standard isotropic $\alpha$-stable Lévy process, and reduces to $-\Delta$ when $\alpha = 2$ (see [10]).

In this thesis, we work mainly in the one-dimensional case, i.e., on $\mathbb{R}$. Hence, (1.1) becomes

$$(-\Delta)^{\alpha/2}u(x) = c_\alpha \int_{-\infty}^{\infty} \frac{u(x) - u(y)}{|x-y|^{1+\alpha}}\, dy, \tag{1.9}$$

where $\alpha \in (0, 2)$, and

$$c_\alpha \equiv c_{1,\alpha} = \frac{2^\alpha \Gamma(\frac{1+\alpha}{2})}{\sqrt{\pi}|\Gamma(-\frac{\alpha}{2})|}.$$

Moreover, we work with the equivalent definition of (1.9):

$$(-\Delta)^{\alpha/2}u(x) = c_\alpha \int_{-\infty}^{\infty} \frac{u(x) - u(x+y)}{|y|^{1+\alpha}} dy. \tag{1.10}$$

In recent years, there has been an increasing interest in evolution equations that incorporate nonlocal operators and, in particular, nonlocal operators that resemble a fractional power of the Laplacian or derivatives of fractional order. There are many models where such operators appear, and there is also an intrinsic mathematical interest in analyzing and simulating such equations. For instance, a discussion on applications where sub- and superdiffusion models might be better suited than classical diffusion ones can be found in the review manuscripts [20–24]. Among others, these include applications in acoustics [18, 25], and quantum mechanics [26]. Lévy flights are a particular example of random motion of particles that leads to superdiffusion and that can be characterized in the macroscopic scale as a diffusion equation by a fractional Laplacian (see, e.g., [27, 28] and the references therein). For a mathematical interpretation of the fractional Laplacian, we refer the reader to, e.g., the monographs [10, 11]. On the other hand, models with fractional diffusion where front propagating solutions ensue have been subject to mathematical analysis (see, e.g., [29–41], to mention a few recent works in this direction).

In this thesis, we concentrate in the example model where the fractional Laplacian appears instead of the usual term of Brownian diffusion. We will simulate as a test a nonlinear reaction-diffusion equation, namely, the following nonlocal Fisher's equation:

$$\partial_t u + (-\Delta)^{\alpha/2} u = f(u), \quad x \in \mathbb{R}, \ t \geq 0, \tag{1.11}$$

where, generically, $f(u) = u(1-u)$ is the so-called monostable nonlinearity; and $f(u) = u(1-u)(u-a)$, with $a \in (0,1)$, is the bistable nonlinearity. With classical diffusion, this is a paradigm equation for pattern forming systems and reaction-diffusion systems in general (see the classical references for the monostable case [5, 42–44], etc., and for the biestable case, [45–49], etc.). The non-local version (1.11) has been proposed as a reaction-diffusion system with anomalous diffusion [30, 50, 51]. Some fundamental analytical results appear in [52, 53], for more general nonlinear equations and in several dimensions. Our main interest here is to simulate (1.11) as an illustration of a problem that requires a very large spatial domain or the whole domain, when traveling wave solutions ensue, since these travel in one direction and they do so with a wave speed exponentially increasing in time in the monostable case (see [54–56]). In this regard, we will contrast the numerical results with the analytical ones, which include, apart from the already mentioned ones, [33–35] for the bistable case (see also [36, 37] for a nonsymmetric non-local operator of Riesz-Feller type).

Let us now describe some numerical aspects relevant to this thesis. To the best of our knowledge, the use of spectral and pseudospectral methods for approximating the fractional Laplacian is limited to a few instances in the literature: we remark the works [57–59], where, although a truncation of the domain is not explicitly given, the method relays on the approximation of the fractional Laplacian by an operator on a truncated domain.

Most of the fractional models deal with infinite or semi-infinite domains. Our aim is thus to present numerical methods that deal accurately and efficiently with problems posed on $\mathbb{R}$, taking particular care of the numerical treatment of $(-\Delta)^{\alpha/2}$.

One of the main difficulties of approximating numerically (2.5) is the unboundedness of the spatial domain; nevertheless, there is a variety of tools for dealing with problems in infinite or semi-infinite domains. For problems posed on $x \in \mathbb{R}$, we have options like:

i. Sinc functions.

ii. Hermite functions.

iii. Algebraically mapped Chebyshev polynomials.

iv. Exponentially mapped Chebyshev polynomials.

v. Solving the problem on a large but finite interval (domain truncation).

For problems posed on the semi-infinite interval $x \in [0, \infty)$ it is possible to use:

i. Laguerre functions.

ii. Chebyshev polynomials with various mappings.

iii. Domain truncation.

In fact, in the study of unboundedness of the spatial domain, according to Boyd [2], the many possible options for unbounded domains fall into three broad categories:

1. Domain truncation (approximation of $x \in \mathbb{R}$ by $[-L, L]$, with $L \gg 1$.
2. Using basis functions intrinsic to an infinite interval (for example, Hermite functions, sinc functions).
3. Mapping the unbounded interval to a finite interval, followed by application of Chebyshev polynomials or a Fourier series.

Observe that these strategies can be combined; for example, mapping and domain truncation, etc. In this work, we will adopt the third one. It is worth mentioning that mapping is equivalent to creating new basis functions, which are the images of Chebyshev polynomials or Fourier series, and whose natural home is the infinite or semi-infinite interval. An infinite variety of maps is possible; but three relevant classes of mappings are the following ones:

1. Logarithmic maps: $x = \operatorname{arctanh}(\psi), \qquad \psi \in [-1, 1]$.
2. Algebraic maps: $x = L\psi/\sqrt{1-\psi^2}, \qquad \psi \in [-1, 1]$.
3. Exponential maps: $x = \sinh(L\psi), \qquad \psi \in [-\pi, \pi]$.

Here $x \in \mathbb{R}$. The names for theses classes of mappings come from the behavior of $x$, i.e, from how $x$ increases when $\vartheta$ tends to $\pm 1$. In general, an advantage of not truncating the domain is that the boundary conditions can be ignored when the domain of integration is infinite [60], while truncating the domain needs setting artificial boundary conditions [61]. We will concentrate our study on the algebraic map, and specifically on the algebraically mapped Chebyshev polynomials $TB_n(x)$, known as the rational Chebyshev functions.

Chebyshev polynomials are a well known set of orthogonal polynomials and have proved useful for numerical analysis. They are a desired choice, because they can be thought of as a Fourier cosine series in disguise [2, 62] and, as such, allow the use of existing fast Fourier transform (FFT) software [3]. On the other hand, the rational Chebyshev functions are a rescaling of the standard Chebyshev polynomials, so that their domain is the entire real line [60, 62]. In [63], it was shown that rational Chebyshev functions are the most versatile option, because they are a good choice for approximating exponentially decaying functions, and they excel when applied to polynomial decaying solutions. In fact, in some of the numerical test that we present in this work, we consider exponentially decaying initial data for the fractional Laplacian. Remark that the rational Chebyshev functions are closely related to the Christov functions and the Higgins functions [7], also considered in this thesis, and which are very adequate for computing numerically the Hilbert transform of functions in $L^2(\mathbb{R})$ (see for instance [64] and [65]), although they are not sufficient for representing functions that are not in $L^2(\mathbb{R})$.

There is a wide range of publications related to the effectiveness of different methods to solve numerically nonlocal fractional operators. In the particular case of the fractional Laplacian, several numerical methods have been proposed recently for

equations associated with this operator in its different equivalent definitions [1]. For instance, either the spectral definition in the Fourier space, or the singular integral representation, which we use in this thesis. Note that most of the numerical evaluations have been developed on bounded domains; for instance, one of the main contributions is the work by Huang and Oberman [66, 67]; where, considering one-dimensional problems, they derived a scheme based on singularity subtraction and finite-difference approximation with a quadrature rule in a bounded domain. Remark that, for smooth enough functions, the accuracy of their method is very close to $\mathcal{O}(h^{3-\alpha})$, where $h$ is the linear spacing between nodes, being up to $10^{-3}$. They also established that the majority of the numerical methods for solving equations involving the fractional Laplacian operator are related to fractional derivatives (either of the Riemann-Liouville type or of the Caputo type) on bounded domains, called fractional diffusion.

In [66], the authors take as reference [68], which summarizes the existing algorithms to solve the fractional derivatives operator, among which we can find the product integration technique based on the trapezoidal quadrature rule, and methods of the type Predict-Evaluate-Correct-Evaluate to solve Caputo-type fractional diffusion equations (FDEs). When used in the discretization of FDEs, the resulting schemes can usually be interpreted as a random walk with long range jumps [69, 70]. Another popular class of schemes use spectral decomposition in the Fourier space or other basis [59], usually valid only for finite spatial domains; note that, in [59], the errors are close to $10^{-7}$. Similar schemes using a finite difference approximation are [71–73].

Taking into account [66, 67], Minden and Ying [74] presented a simple solver for the fractional Laplacian in multiple dimensions based on the hypersingular integral representation. Through singularity subtraction, they obtained a regularized integrand to which they apply the trapezoidal rule. When the function $u$ is sufficiently smooth, standard results on convergence of the trapezoidal rule and finite-difference operators imply that the computation errors tend to zero as $\mathcal{O}(h^2)$. The work [74] offers a vast number of references on numerical solutions of one-dimensional fractional Laplacian systems using fast preconditioned iterative methods; for instance, [75–78].

In the same line of work, but separately, Duo and Zhang [79] proposed a method that provide a fractional analogue of the central difference schemes for the fractional Laplacian. However, as in [74], they considered bounded domains for the two- and three- dimensional cases with extended homogeneous boundary conditions. They also achieved errors tending to zero as $\mathcal{O}(h^2)$, for any $\alpha \in (0, 2)$, and their results hold for dimensions two and three. In [79], we can find various references on the finite element method (FEM) to solve problems involving the fractional Laplacian operator, and numerical results are presented for one-, two-, and three-dimensional cases (see [80–83]). Some works using finite-element-based approaches are [84–86], and some general references for the approximation of the fractional Laplacian on bounded domains include [87–89].

There are notable algorithms for discretizing the fractional Laplacian based on the

Caffarelli-Silvestre extension [19, 90, 91], followed by the application of spectral approaches [85, 86, 92]. A more recent publication is, for instance, [93], where a fast spectral Galerkin method using the generalized Laguerre functions for the extension problem is applied. The convergence rate with respect to $h$ is essentially of second order, which is the expected rate for the $L^2$-norm of the error when using piecewise linear finite elements, until the dominating errors are those in the extended domain. Moreover, Chen et al. [94] presented a solver via multilevel techniques to deal with extension problems (see [95–97], for some other related works).

All these publications have in common the use of truncation in the integration domain, or that they solve problems on a bounded domain, where different definitions of the fractional Laplacian are not equivalent. Therefore, in general, truncation of the domain leads to a natural deterioration of the rate of convergence. On the other hand, the accuracy of these methods is well below spectral accuracy, which we have virtually achieved in this thesis in dimension one; more precisely, we have obtained errors close to the epsilon of the machine for smooth enough functions.

The structure of this thesis is as follows. In Chapter 2 (cf. [98]), we present a new pseudospectral method to compute efficiently and accurately the fractional Laplacian (1.10) without using truncation of the domain. We start by giving a new expression of the Laplacian in a more suitable way, which requires at least $\mathcal{C}^2$ regularity. Then, we map the unbounded interval $\mathbb{R}$ into a finite one $[0, \pi]$ by using an algebraic map given by the change of variable $x = L\cot(s)$, with $L > 0$ being a scaling factor, and $s \in [0, \pi]$. We expand $u(s) \equiv u(L\cot(s))$ into a Fourier series, and, at its turn, obtain the Fourier series expansion of $(-\Delta)^{\alpha/2}(e^{iks})$, which constitutes the central part of this chapter. We also show the construction of an operational matrix $\mathbf{M}_\alpha$ that multiply the column vector formed by the Fourier coefficients of $u(s)$ to approximate $(-\Delta)^{\alpha/2}(u(s))$ at the non-final nodes $s_j = \pi(2j+1)/(2N)$, with $0 \leq j \leq N-1$. In addition, we develop an error analysis to justify the convergence of the method; and, in order to measure the accuracy of our scheme, we test it with a couple of functions for which an explicit expression of the fractional Laplacian is known. Later on, as an application ot the method, we simulate numerically the evolution of (1.11). Finally, we point out futures lines or research; in particular, we offer an alternative formula to (1.1) for the fractional Laplacian in $\mathbb{R}^n$, which could serve as a starting point.

In Chapter 3 (cf. [99]), using complex variable techniques, we obtain explicit expressions, in terms of the ${}_2F_1$ Gaussian hypergeometric function, for the one-dimensional fractional Laplacian of the Higgins and Christov functions, and of their sine-like and cosine-like versions. Then, we also explain how to implement these expressions efficiently in MATLAB [8], and give numerical examples as an application. Furthermore, we test their adequacy from a numerical point of view, comparing the numerical results with those in Chapter 2 (and thus confirming the correctness of the method presented in Chapter 2). On the one hand, after discussing the numerical difficulties in the implementation of the proposed formulas, for moderately large values of $n$, we develop a method using variable precision arithmetic that gives accurate results; on the other hand, our implementation of ${}_2F_1$ largely outperforms that of MATLAB. After that, even if the method in Chapter 2 is faster, the method in this chapter is much easier to implement and still competitive for not too large values of $n$.

In Chapter 4, using some ideas from [9], we discuss other approaches to approximate numerically the fractional Laplacian on $\mathbb{R}$, using fast convolution. More precisely, we apply again the mapping $x = L\cot(s)$, to obtain a singular integral on $s \in [0, \pi]$, and try different quadrature formulas, expressing them as discrete convolutions. At this point, we apply the discrete convolution theorem, that states that the discrete Fourier transform of a discrete convolution equals the product of the discrete Fourier transforms, namely, $(\widehat{u * v})_p = \widehat{u}_p \widehat{v}_p$, where $u = \{u_r\}$ and $v = \{v_r\}$ are two sequences of numbers. Since we use the fast Fourier Transform (FFT) [3] to compute the discrete Fourier transforms, this technique can be referred to as fast convolution, because it requires $\mathcal{O}(N \log N)$ operations when the two sequences $u$ and $v$ are $N$-periodic; this is an important reduction in the number of operations of a direct summation procedure, which requires $\mathcal{O}(N^2)$ operations (and, hence, becomes expensive for relative small values of $N$). Following the previous arguments, we develop a method that appears to be of second-order for all $\alpha \in (0, 1) \cup (1, 2)$, and that, even if less accurate than those exposed in Chapters 2 and 3, is exceedingly fast. We test it for the functions considered in Chapter 2 and show how Richardson's extrapolation can be used to improve the results. To finish this chapter, we adapt the numerical method in [100] to the computation of the fractional Laplacian, rewriting it so the fast convolution can also be applied to it, and implement again Richardson's extrapolation to improve the results, too.

To close this thesis, we have added Appendix A, where we list some important properties of the Chebyshev polynomials, and of the rational Chebyshev functions defined on $\mathbb{R}$.

# Chapter 2

# Computation of the fractional Laplacian for regular functions

The structure of this chapter is as follows. In Section 2.1, we propose a novel method to compute accurately the fractional Laplacian (1.10) without using truncation. More precisely, we rewrite the fractional Laplacian in a more suitable way, which requires at least $\mathcal{C}^2$ regularity; then, after mapping the original domain $\mathbb{R}$ to $[0, \pi]$ by using the change of variable $x = L\cot(s)$, with $L > 0$, $s \in [0, \pi]$, we expand $u(s) \equiv u(L\cot(s))$ in Fourier series, and, at its turn, obtain the Fourier series expansion of $(-\Delta)^{\alpha/2}(e^{iks})$, which constitutes the central part of this chapter. We also show how to generate efficiently an operational matrix $\mathbf{M}_\alpha$ that can be applied to the coefficients of the Fourier expansion of $u(s)$, to approximate $(-\Delta)^{\alpha/2}(u(s))$ at the equally-spaced nodes

$$s_j = \frac{\pi(2j+1)}{2N}, \quad 0 \leq j \leq N-1. \tag{2.1}$$

Furthermore, we justify theoretically the convergence of the method. Later on, in Section 2.2, we test the proposed method for a couple of functions; and, in Section 2.3, we apply it to the numerical simulation of (1.11) in the monostable case.

To the best of our knowledge, the numerical computation of the fractional Laplacian without truncating the domain has not being done so far. However, the change of variable $x = L\cos(s)$ was applied successfully in [100] to compute numerically a related nonlocal operator defined on the whole real line. More precisely, in [100], from which we get several useful ideas, $\partial_x \mathcal{D}^\alpha$ was considered on $\mathbb{R}$, where the operator $\mathcal{D}^\alpha$ can be regarded as a left-sided fractional derivative in the Caputo sense (see, e.g., [101]), with integration taken from $-\infty$:

$$\mathcal{D}^\alpha u(x) = \frac{1}{\Gamma(1-\alpha)} \int_{-\infty}^{x} \frac{u_x(y)}{(x-y)^\alpha} dy, \quad \alpha \in (0,1). \tag{2.2}$$

After defining $u(s) \equiv u(L\cot(s))$, $\partial_x \mathcal{D}^\alpha(u(s))$ was approximated at the nodes $s_j$ in (2.1) by the composite midpoint rule taken over the families of nodes

$$s_l^{(m)} = \frac{\pi(2l+1)}{2^{m+1}N}, \quad 0 \leq l \leq 2^m N - 1, \quad m = 1, 2, \ldots, \tag{2.3}$$

although, in practice, only the indices $l$ satisfying $2^{m-1}(2j+1) \leq l \leq 2^m N - 1$ were used, denoting as $[\partial_x \mathcal{D}^\alpha]^{(m)}(u(s))$ the resulting approximation. Then, studying the errors of several functions with different types of decay and applying Richardson extrapolation [102] to $[\partial_x \mathcal{D}^\alpha]^m(u(s))$, it was conjectured that

$$\begin{aligned} &\|[\partial_x \mathcal{D}^\alpha]^{(m)} u(x) - \partial_x \mathcal{D}^\alpha u(x)\|_\infty \\ &\qquad = \frac{c_1(\alpha)}{m^{2-\alpha}} + \frac{c_2(\alpha)}{m^{3-\alpha}} + \frac{c_3(\alpha)}{m^{4-\alpha}} + \frac{c_4(\alpha)}{m^{5-\alpha}} + \frac{c_5(\alpha)}{m^{6-\alpha}} + \dots, \end{aligned} \tag{2.4}$$

and, indeed, this formula yielded very accurate results, at least for the functions considered. Remark that, in practice, $u(s)$ was expanded in Fourier series, so the extrapolation was really applied over $[\partial_x \mathcal{D}^\alpha]^{(m)}(e^{iks})$, which enabled to create an operational matrix acting on the coefficients of the Fourier expansion of $u(s)$.

As we can see, the main difference between the method developed in this chapter and [100] is the numerical computation of the corresponding nonlocal operator acting on a single Fourier mode $e^{iks}$. In this chapter, we have not considered the extrapolation technique, because it appears to be more involved than in [100] (we will comment on this in Section 4.5), and, on the other hand, the method that we are proposing here is, in our opinion, very accurate.

To the best of our knowledge, the use of spectral and pseudospectral methods for approximating the fractional Laplacian is limited to a few instances in the literature: we remark the works [57–59], where, although a truncation of the domain is not explicitly given, the method relays on the approximation of the fractional Laplacian by an operator on a truncated domain.

In the following pages, we will use the representation of (1.10), together with Lemma 2.1.1, which requires boundedness and $\mathcal{C}^2$-regularity.

## 2.1 Computation of the fractional Laplacian for regular functions

In the following pages, we will develop a new method to approximate numerically (1.10). However, instead of working directly with (1.10), we will use the representation given by the following lemma, which requires boundedness and $\mathcal{C}^2$-regularity.

### 2.1.1 Equivalent form of the fractional Laplacian for regular functions

**Lemma 2.1.1.** *Consider the twice continuous bounded function $u \in \mathcal{C}_b^2(\mathbb{R})$. If $\alpha \in [1,2)$, or $\alpha \in (0,1)$ and $\lim_{x\to\pm\infty} u_x(x) = 0$, then*

$$(-\Delta)^{\alpha/2} u(x) = \begin{cases} \dfrac{1}{\pi} \displaystyle\int_{-\infty}^{\infty} \frac{u_x(y)}{x-y} dy, & \alpha = 1, \\ \dfrac{c_\alpha}{\alpha(1-\alpha)} \displaystyle\int_{-\infty}^{\infty} \frac{u_{xx}(y)}{|x-y|^{\alpha-1}} dy, & \alpha \neq 1. \end{cases} \tag{2.5}$$

*Proof.* Let us express first (1.10) as an integral over $[0,\,\infty)$:

$$\begin{aligned}(-\Delta)^{\alpha/2}u(x) &= c_\alpha \int_0^\infty \frac{u(x)-u(x-y)+u(x)-u(x+y)}{y^{1+\alpha}}dy\\ &= c_\alpha \int_0^\infty \int_0^y \frac{u_x(x-z)-u_x(x+z)}{y^{1+\alpha}}dz\,dy\\ &= c_\alpha \int_0^\infty \left[(u_x(x-z)-u_x(x+z))\int_z^\infty \frac{1}{y^{1+\alpha}}dy\right]dz\\ &= \frac{c_\alpha}{\alpha}\int_0^\infty \frac{u_x(x-z)-u_x(x+z)}{z^\alpha}dz, \end{aligned} \tag{2.6}$$

where we have changed the order of integration; observe that this is equivalent to

$$\begin{aligned}(-\Delta)^{\alpha/2}u(x) &= \frac{c_\alpha}{\alpha}\int_{-\infty}^\infty \frac{y\,u_x(x-y)}{|y|^{1+\alpha}}dy\\ &= -\frac{c_\alpha}{\alpha}\int_{-\infty}^\infty \frac{y\,u_x(x+y)}{|y|^{1+\alpha}}dy. \end{aligned} \tag{2.7}$$

We distinguish three cases. When $\alpha = 1$, $c_1 = 1/\pi$, so

$$\begin{aligned}(-\Delta)^{1/2}u(x) &= \frac{1}{\pi}\int_0^\infty \frac{u_x(x-y)-u_x(x+y)}{y}dy\\ &= \frac{1}{\pi}\int_{-\infty}^\infty \frac{u_x(y)}{x-y}dy, \end{aligned}$$

i.e., $(-\Delta)^{1/2}u(x)$ is precisely the Hilbert transform [12] of $u_x(x)$, as mentioned in the introduction. On the other hand, when $\alpha \in (1,2)$,

$$\begin{aligned}(-\Delta)^{\alpha/2}u(x) &= \frac{c_\alpha}{\alpha}\int_0^\infty \frac{u_x(x-z)-u_x(x)+u_x(x)-u_x(x+z)}{z^\alpha}dz\\ &= -\frac{c_\alpha}{\alpha}\int_0^\infty \int_0^z \frac{u_{xx}(x-y)+u_{xx}(x+y)}{z^\alpha}dy\,dz\\ &= -\frac{c_\alpha}{\alpha}\int_0^\infty \left[(u_{xx}(x-y)+u_{xx}(x+y))\int_y^\infty \frac{1}{z^\alpha}dz\right]dy\\ &= -\frac{c_\alpha}{\alpha(\alpha-1)}\int_0^\infty \frac{u_{xx}(x-y)+u_{xx}(x+y)}{y^{\alpha-1}}dy\\ &= \frac{c_\alpha}{\alpha(1-\alpha)}\int_{-\infty}^\infty \frac{u_{xx}(x+y)}{|y|^{\alpha-1}}dy\\ &= \frac{c_\alpha}{\alpha(1-\alpha)}\int_{-\infty}^\infty \frac{u_{xx}(y)}{|x-y|^{\alpha-1}}dy. \end{aligned}$$

Finally, when $\alpha \in (0,1)$, this last formula also holds, although the deduction is slightly different, and $\lim_{x\to\pm\infty} u_x(x) = 0$ is required. Indeed, from (2.6),

$$\begin{aligned}(-\Delta)^{\alpha/2}u(x) &= \frac{c_\alpha}{\alpha}\int_0^\infty\int_z^\infty \frac{u_{xx}(x-y)+u_{xx}(x+y)}{z^\alpha}dy\,dz\\ &= \frac{c_\alpha}{\alpha}\int_0^\infty\left[(u_{xx}(x-y)+u_{xx}(x+y))\int_0^y \frac{1}{z^\alpha}dz\right]dy\\ &= \frac{c_\alpha}{\alpha(1-\alpha)}\int_0^\infty \frac{u_{xx}(x-y)+u_{xx}(x+y)}{y^{\alpha-1}}dy\\ &= \frac{c_\alpha}{\alpha(1-\alpha)}\int_{-\infty}^\infty \frac{u_{xx}(y)}{|x-y|^{\alpha-1}}dy,\end{aligned}$$

which completes the proof of the lemma. □

### 2.1.2 Mapping $\mathbb{R}$ to a finite interval

Among the different possible mappings which we have mentioned in the introduction, we use the algebraic one. More precisely, if we consider $x \in \mathbb{R}$, $\psi \in [-1,1]$, and $s \in [0,\pi]$, then the different domains are mapped onto one another through the following relationships [2]:

$$\psi = \frac{x}{\sqrt{L^2+x^2}} \Longleftrightarrow x = \frac{L\psi}{\sqrt{1-\psi^2}},$$

and

$$x = L\cot(s) \Longleftrightarrow s = \operatorname{arccot}\left(\frac{x}{L}\right),$$

for a certain constant $L > 0$. Small values of $L$ cause the majority of the discretized points in the abscissa to be clumped near the origin, whereas increasing $L$ pulls out the points from the origin toward the rest of the domain. The $n$th rational Chebyshev is

$$TB_n(x) \equiv T_n(\psi) \equiv \cos(nt), \quad \forall n \in \mathbb{N}\cup\{0\}.$$

i.e.,

$$TB_n(x) = T_n\left(\frac{x}{\sqrt{L^2+x^2}}\right) = \cos\left(n\operatorname{arccot}\left(\frac{x}{L}\right)\right), \quad \forall n \in \mathbb{N}\cup\{0\}, \tag{2.8}$$

where $T_n(\psi)$ is the standard Chebyshev polynomials and $TB_n(x)$ are the rational Chebyshev functions, which form an orthogonal basis in $\mathbb{R}$ with respect to the weight $(1+x^2)^{-1}$:

$$\int_{-\infty}^\infty \frac{TB_m(x)TB_n(x)}{1+x^2}\,dx = \begin{cases}\frac{\pi}{2}, & m=n>0,\\ \pi, & m=n=0,\\ 0, & m\neq n.\end{cases}$$

We present more details about the properties of Chebyshev polynomials and rational Chebyshev functions in Appendix A. Although we can work with (2.8), we prefer to use a trigonometric representation on them, which allows us an easier handling of them. Thus, we are going to consider the following change of variable:

$$\psi = \cos(s) \Longleftrightarrow x = L\cot(s), \quad s \in [0, \pi]. \tag{2.9}$$

Then, (2.8) becomes

$$T_n(\psi) = TB_n(x) = \cos(ns).$$

Therefore, a series expansion in terms of Chebyshev polynomials or rational Chebyshev functions is equivalent to a cosine expansion.

In this thesis, we work with (2.7) rather than with (2.5), because the calculations in the next sections appear to be simpler. In order to express (2.5) in terms of $s \in [0, \pi]$, we need the following identities [60]:

$$\begin{aligned} u_x(x) &= -\frac{\sin^2(s)}{L} u_s(s), \\ u_{xx}(x) &= \frac{\sin^4(s)}{L^2} u_{ss}(s) + \frac{2\sin^3(s)\cos(s)}{L^2} u_s(s), \end{aligned} \tag{2.10}$$

where, with some abuse of notation, $u(s) \equiv u(x(s))$. Then, bearing in mind that $dx = -L\sin^{-2}(s)ds$, (2.5) becomes

$$(-\Delta)^{\alpha/2} u(s) = \begin{cases} -\dfrac{1}{L\pi} \displaystyle\int_0^\pi \frac{u_s(\eta)}{\cot(s) - \cot(\eta)} d\eta, & \alpha = 1, \\ \dfrac{c_\alpha}{L^\alpha \alpha(1-\alpha)} & \\ \cdot \displaystyle\int_0^\pi \frac{\sin^2(\eta) u_{ss}(\eta) + 2\sin(\eta)\cos(\eta) u_s(\eta)}{|\cot(s) - \cot(\eta)|^{\alpha-1}} d\eta, & \alpha \neq 1, \end{cases} \tag{2.11}$$

or, equivalently,

$$(-\Delta)^{\alpha/2} u(s) = \begin{cases} \dfrac{\sin(s)}{L\pi} \displaystyle\int_0^\pi \frac{\sin(\eta) u_s(\eta)}{\sin(s-\eta)} d\eta, & \alpha = 1, \\ \dfrac{c_\alpha |\sin(s)|^{\alpha-1}}{L^\alpha \alpha(1-\alpha)} & \\ \cdot \displaystyle\int_0^\pi \frac{\sin^\alpha(\eta)(\sin(\eta) u_{ss}(\eta) + 2\cos(\eta) u_s(\eta))}{|\sin(s-\eta)|^{\alpha-1}} d\eta, & \alpha \neq 1. \end{cases} \tag{2.12}$$

### 2.1.3 Discretizing the mapped bounded domain

We discretize the interval $s \in [0, \pi]$ in the nodes defined in (2.1):

$$s_j = \frac{\pi(2j+1)}{2N} \Longleftrightarrow x_j = L\cot\left(\frac{\pi(2j+1)}{2N}\right) \Longleftrightarrow \psi_j = \cos\left(\frac{\pi(2j+1)}{2N}\right),$$

such that $s_0 = \pi/(2N)$, $s_{N-1} = \pi - \pi/(2N)$, $s_{j+1} - s_j = \pi/N$, where $0 \leq j \leq N-1$, which divide the interval $[0, \pi]$ in $N$ equally-spaced parts. Therefore, we avoid evaluating (2.11) directly at $s = 0$ and $s = \pi$. Note that it is also possible to introduce a spacial shift in $x$, i.e.,

$$x = x_c + L\cot(s); \tag{2.13}$$

so

$$x_j = x_c + L\cot(s_j) = x_c + L\cot\left(\frac{\pi(2j+1)}{2N}\right).$$

In general, along this thesis, whenever we write $u(x_j)$, we refer to $u(x)$ evaluated at $x = x_j$, whereas, we write $u(s_j)$ to refer to $u(x(s))$ evaluated at $s_j$. Therefore, with some abuse of notation, $u(s_j) \equiv u(x_j)$. Observe that the definition of $s_j$ in (2.1) does not depend on $x_c$, whereas the definition of $x_j$ does, which makes it preferable to work with $u(s_j)$. On the other hand, as mentioned above, since $s \in [0, \pi]$, a cosine series expansion is enough to represent $u(s)$. However, we rather consider a more general series expansion formed by $e^{iks}$, with $k \in \mathbb{Z}$, which is somehow easier to implement numerically:

$$u(s) = \sum_{k=-\infty}^{\infty} \hat{u}(k) e^{iks}, \quad s \in [0, \pi].$$

Hence, in order to determine the coefficients $\hat{u}(k)$, we have to extend the definition of $u(s)$ to $s \in [0, 2\pi]$. Therefore, we discretize $[0, \pi]$ in $2N$ nodes, $s_j = \pi(2j+1)/(2N)$, $0 \leq j \leq 2N-1$. Note that an even expansion of $u(s)$ at $s = \pi$ will yield precisely a cosine series, while an odd extension will yield a sine series.

Since it is impossible to work with infinitely many frequencies, we approximate $u(s)$ as

$$u(s) \approx \sum_{k=-N}^{N-1} \hat{u}(k) e^{iks}, \quad s \in [0,\ 2\pi], \tag{2.14}$$

which is a $2N$-term approximation of a $2\pi$-periodic function $u$.

Then, taking $0 \leq j \leq 2N-1$ in (2.1), we adopt a pseudospectral approach to determine uniquely the $2N$ coefficients $\hat{u}(k)$ in (2.14), i.e., we impose (2.14) to be an equality at $s_j$:

$$\begin{aligned} u(s_j) \equiv \sum_{k=-N}^{N-1} \hat{u}(k) e^{iks_j} &= \sum_{k=-N}^{N-1} \hat{u}(k) e^{ik\pi(2j+1)/(2N)} \\ &= \sum_{k=0}^{N-1} \left[\hat{u}(k) e^{ik\pi/(2N)}\right] e^{2ijk\pi/(2N)} \\ &\quad + \sum_{k=N}^{2N-1} \left[\hat{u}(k-2N) e^{i(k-2N)\pi/(2N)}\right] e^{2ijk\pi/(2N)}. \end{aligned} \tag{2.15}$$

Equivalently, the coefficients $\hat{u}(k)$ are given by

$$\hat{u}(k) \equiv \frac{e^{-ik\pi/(2N)}}{2N} \sum_{j=0}^{2N-1} u(s_j) e^{-2ijk\pi/(2N)}. \tag{2.16}$$

Note that the discrete Fourier transforms (2.15) and (2.16) can be computed very efficiently by means of the fast Fourier Transform (FFT) [3]. On the other hand, we apply systematically a Krasny filter [103], i.e., we set to zero all the Fourier coefficients $\hat{u}(k)$ with modulus smaller than a fixed threshold, which in this thesis is the epsilon of the machine.

### 2.1.4 An explicit calculation of $(-\Delta)^{\alpha/2} e^{iks}$

Since we are approximating $u(s)$ by (2.14), the problem is reduced to computing $(-\Delta)^{\alpha/2} e^{iks}$. In this section, we will prove the following theorem:

**Theorem 2.1.2.** *Let $\alpha \in (0,1) \cup (1,2)$, then*

$$(-\Delta)^{\alpha/2}(e^{iks}) = \begin{cases} \dfrac{c_\alpha |\sin(s)|^{\alpha-1}}{8L^\alpha \tan(\frac{\pi\alpha}{2})} \displaystyle\sum_{l=-\infty}^{\infty} e^{i2ls}((1-\alpha)k^2 - 4kl) \\ \quad \cdot \dfrac{\Gamma\left(\frac{-1+\alpha}{2} + |l|\right)\Gamma\left(\frac{-1-\alpha}{2} + \left|\frac{k}{2} - l\right|\right)}{\Gamma\left(\frac{3-\alpha}{2} + |l|\right)\Gamma\left(\frac{3+\alpha}{2} + \left|\frac{k}{2} - l\right|\right)}, & k \text{ even}, \\ i\dfrac{c_\alpha |\sin(s)|^{\alpha-1}}{8L^\alpha} \displaystyle\sum_{l=-\infty}^{\infty} e^{i2ls}((1-\alpha)k^2 - 4kl) \\ \quad \cdot \operatorname{sgn}(\frac{k}{2} - l) \dfrac{\Gamma\left(\frac{-1+\alpha}{2} + |l|\right)\Gamma\left(\frac{-1-\alpha}{2} + \left|\frac{k}{2} - l\right|\right)}{\Gamma\left(\frac{3-\alpha}{2} + |l|\right)\Gamma\left(\frac{3+\alpha}{2} + \left|\frac{k}{2} - l\right|\right)}, & k \text{ odd}. \end{cases} \tag{2.17}$$

*Moreover, when $\alpha = 1$,*

$$(-\Delta)^{1/2}(e^{iks}) = \begin{cases} \dfrac{|k|\sin^2(s)}{L} e^{iks}, & k \text{ even}, \\ \dfrac{ik}{L\pi}\left(\dfrac{-2}{k^2-4} - \displaystyle\sum_{l=-\infty}^{\infty} \frac{4\operatorname{sgn}(l) e^{i2ls}}{(k-2l)((k-2l)^2-4)}\right), & k \text{ odd}. \end{cases} \tag{2.18}$$

*Proof.* We prove first the case $\alpha = 1$. Introducing $u(s) = e^{iks}$ in (2.12), we get

$$(-\Delta)^{1/2}(e^{iks}) = \frac{ik\sin(s)}{L\pi} \int_0^\pi \frac{\sin(\eta) e^{ik\eta}}{\sin(s-\eta)} d\eta. \tag{2.19}$$

When $k \equiv 0 \bmod 2$,

$$\int_0^\pi \frac{\sin(\eta)e^{ik\eta}}{\sin(s-\eta)}d\eta = -e^{iks}\cos(s)\int_0^\pi e^{ik\eta}d\eta - e^{iks}\sin(s)\int_0^\pi \frac{\cos(\eta)e^{ik\eta}}{\sin(\eta)}d\eta.$$

The first integral is trivial, and the second can be calculated explicitly, too:

$$\begin{aligned}\int_0^\pi \frac{\cos(\eta)e^{ik\eta}}{\sin(\eta)}d\eta &= \frac{i\operatorname{sgn}(k)}{2}\int_0^{2\pi}\frac{\cos(\eta)\sin(|k|\eta)}{\sin(\eta)}d\eta \\ &= \begin{cases} 0, & k=0, \\ i\pi\operatorname{sgn}(k), & k\in 2\mathbb{Z}\backslash\{0\}; \end{cases}\end{aligned}$$

which is easily proved by induction on $2\mathbb{N}$, bearing into account that $\sin(2\eta) = 2\sin(\eta)\cos(\eta)$, and that $\sin((|k|+2)\eta) - \sin(|k|\eta) = 2\sin(\eta)\cos((|k|+1)\eta)$. Therefore,

$$\int_0^\pi \frac{\sin(\eta)e^{ik\eta}}{\sin(s-\eta)}d\eta = \begin{cases} -\pi\cos(s), & k=0, \\ -i\pi\operatorname{sgn}(k)\sin(s)e^{iks}, & k\in 2\mathbb{Z}\backslash\{0\}; \end{cases} \tag{2.20}$$

from which follows the first part of (2.18). On the other hand, when $k \equiv 1 \bmod 2$, $e^{ik\eta}$ is not periodic in $\eta \in [0,\pi]$, and there seems to be no compact formula for $(-\Delta)^{1/2}(e^{iks})$, as in $k \equiv 0 \bmod 2$. Hence, we have to consider a series expansion for $(-\Delta)^{1/2}(e^{iks})$; more precisely, we write

$$\sin(s)\int_0^\pi \frac{\sin(\eta)e^{ik\eta}}{\sin(s-\eta)}d\eta = \sum_{l=-\infty}^{\infty} c_{kl}e^{i2ls}, \tag{2.21}$$

with $c_{kl}$ given by

$$\begin{aligned} c_{kl} &= \frac{1}{\pi}\int_0^\pi \left[\sin(s)\int_0^\pi \frac{\sin(\eta)e^{ik\eta}}{\sin(s-\eta)}d\eta\right]e^{-i2ls}ds \\ &= \frac{1}{\pi}\int_0^\pi \sin(\eta)e^{ik\eta}\left[\int_0^\pi \frac{\sin(s)e^{-i2ls}}{\sin(s-\eta)}ds\right]d\eta, \end{aligned}$$

where we have changed the order of integration. The inner integral is given by (2.20):

$$\int_0^\pi \frac{\sin(s)e^{-i2ls}}{\sin(s-\eta)}ds = \begin{cases} \pi\cos(\eta), & l=0, \\ -i\pi\operatorname{sgn}(l)\sin(\eta)e^{-i2l\eta}, & l\in\mathbb{Z}\backslash\{0\}. \end{cases}$$

Hence,

$$c_{kl} = \begin{cases} \displaystyle\int_0^\pi \sin(\eta)\cos(\eta)e^{ik\eta}d\eta = \frac{-2}{k^2-4}, & l=0, \\ \displaystyle -i\operatorname{sgn}(l)\int_0^\pi \sin^2(\eta)e^{i(k-2l)\eta}d\eta = \frac{-4\operatorname{sgn}(l)}{(k-2l)((k-2l)^2-4)}, & l\neq 0, \end{cases} \tag{2.22}$$

from which we conclude the second part of (2.18).

We consider now $\alpha \neq 1$. Introducing $u(s) = e^{iks}$ in (2.12), we get

$$(-\Delta)^{\alpha/2}(e^{iks}) = \frac{c_\alpha |\sin(s)|^{\alpha-1}}{L^\alpha \alpha(1-\alpha)} \int_0^\pi \frac{\sin^\alpha(\eta)(-k^2\sin(\eta) + 2ik\cos(\eta))e^{ik\eta}}{|\sin(s-\eta)|^{\alpha-1}} d\eta. \quad (2.23)$$

Then, as in (2.21), we consider a series expansion:

$$\int_0^\pi \frac{\sin^\alpha(\eta)(-k^2\sin(\eta) + 2ik\cos(\eta))e^{ik\eta}}{|\sin(s-\eta)|^{\alpha-1}} d\eta = \sum_{l=-\infty}^{\infty} d_{kl} e^{i2ls}, \quad (2.24)$$

with $d_{kl}$ given by

$$\begin{aligned} d_{kl} &= \frac{1}{\pi} \int_0^\pi \left[ \int_0^\pi \frac{\sin^\alpha(\eta)(-k^2\sin(\eta) + 2ik\cos(\eta))e^{ik\eta}}{|\sin(s-\eta)|^{\alpha-1}} d\eta \right] e^{-i2ls} ds \\ &= \frac{1}{\pi} \int_0^\pi \sin^\alpha(\eta)(-k^2\sin(\eta) + 2ik\cos(\eta))e^{ik\eta} \left[ \int_0^\pi \frac{e^{-i2ls}}{|\sin(s-\eta)|^{\alpha-1}} ds \right] d\eta \\ &= \frac{1}{\pi} \left[ \int_0^\pi \frac{e^{-i2ls}}{\sin^{\alpha-1}(s)} ds \right] \left[ \int_0^\pi \sin^\alpha(\eta)(-k^2\sin(\eta) + 2ik\cos(\eta))e^{i(k-2l)\eta} d\eta \right] \\ &= \frac{1}{\pi} I_1 \cdot I_2, \end{aligned} \quad (2.25)$$

where we have changed again the order of integration. Integrals of the type of $I_1$ and $I_2$ can be explicitly calculated by means of standard complex-variable techniques (see, e.g., [104, p. 158], for a classic reference). On the other hand, we have used MATHEMATICA [105], which computes them immediately (after, occasionally, very minor rewriting). The expression for $I_1$ is

$$\begin{aligned} I_1 &= \frac{e^{-i2\pi l}((2i)^\alpha + (-2i)^\alpha e^{i2\pi l})\pi \csc(\pi\alpha)\Gamma\left(\frac{-1+\alpha-2l}{2}\right)}{4\Gamma(-1+\alpha)\Gamma\left(\frac{3-\alpha-2l}{2}\right)} \\ &= -\frac{2^{\alpha-1}\cos(\frac{\pi\alpha}{2})\Gamma(2-\alpha)\Gamma\left(\frac{-1+\alpha}{2} - l\right)}{\Gamma\left(\frac{3-\alpha}{2} - l\right)}, \end{aligned} \quad (2.26)$$

where we have used the well-known Euler's reflection formula:

$$\Gamma(z)\Gamma(1-z) = \frac{\pi}{\sin(\pi z)}.$$

Moreover, applying twice Euler's reflection formula,

$$\begin{aligned}\frac{\Gamma(z)}{\Gamma(w)} &= \frac{\Gamma(z)\Gamma(1-z)\Gamma(1-w)}{\Gamma(w)\Gamma(1-w)\Gamma(1-z)} \\ &= \frac{\sin(\pi w)}{\sin(\pi z)}\frac{\Gamma(1-w)}{\Gamma(1-z)}.\end{aligned} \tag{2.27}$$

Therefore, for $l \in \mathbb{Z}$,

$$\begin{aligned}\frac{\Gamma\left(\frac{-1+\alpha}{2}-l\right)}{\Gamma\left(\frac{3-\alpha}{2}-l\right)} &= \frac{\sin\left(\pi\left(\frac{3-\alpha}{2}-l\right)\right)}{\sin\left(\pi\left(\frac{-1+\alpha}{2}-l\right)\right)}\frac{\Gamma\left(1-\left(\frac{3-\alpha}{2}-l\right)\right)}{\Gamma\left(1-\left(\frac{-1+\alpha}{2}-l\right)\right)} \\ &= \frac{\Gamma\left(\frac{-1+\alpha}{2}+l\right)}{\Gamma\left(\frac{3-\alpha}{2}+l\right)},\end{aligned} \tag{2.28}$$

so the value of $I_1$ does not depend on the sign of $l$, and we can replace the appearances of $l$ in (2.26) by $-l$, $|l|$ or $-|l|$. In this thesis, we consider the last option, getting

$$I_1 = -\frac{2^{\alpha-1}\cos(\frac{\pi\alpha}{2})\Gamma(2-\alpha)\Gamma\left(\frac{-1+\alpha}{2}+|l|\right)}{\Gamma\left(\frac{3-\alpha}{2}+|l|\right)}, \tag{2.29}$$

which is more convenient from an implementation point of view, as we will explain in Section 2.1.5. Likewise, the expression for $I_2$ is

$$\begin{aligned}I_2 &= -2^{-2-\alpha}e^{-i\pi(\alpha+4l)/2}\left((-1)^k + e^{i\pi(\alpha+2l)}\right) \\ &\quad \cdot \frac{k((-1+\alpha)k+4l)\pi\csc(\pi\alpha)\Gamma\left(\frac{-1-\alpha+k-2l}{2}\right)}{\Gamma(-\alpha)\Gamma\left(\frac{3+\alpha+k-2l}{2}\right)} \\ &= \begin{cases} \dfrac{\pi\alpha(1-\alpha)((-1+\alpha)k^2+4kl)\Gamma\left(\frac{-1-\alpha}{2}+\frac{k}{2}-l\right)}{2^{2+\alpha}\sin(\frac{\pi\alpha}{2})\Gamma(2-\alpha)\Gamma\left(\frac{3+\alpha}{2}+\frac{k}{2}-l\right)}, & k \text{ even}, \\ i\dfrac{\pi\alpha(1-\alpha)((-1+\alpha)k^2+4kl)\Gamma\left(\frac{-1-\alpha}{2}+\frac{k}{2}-l\right)}{2^{2+\alpha}\cos(\frac{\pi\alpha}{2})\Gamma(2-\alpha)\Gamma\left(\frac{3+\alpha}{2}+\frac{k}{2}-l\right)}, & k \text{ odd}. \end{cases}\end{aligned}$$

Then, applying again (2.27), we get expressions similar to (2.28):

$$\frac{\Gamma\left(\frac{-1-\alpha}{2}+\frac{k}{2}-l\right)}{\Gamma\left(\frac{3+\alpha}{2}+\frac{k}{2}-l\right)} = \begin{cases} \dfrac{\Gamma\left(\frac{-1-\alpha}{2}-\left(\frac{k}{2}-l\right)\right)}{\Gamma\left(\frac{3+\alpha}{2}-\left(\frac{k}{2}-l\right)\right)}, & k \text{ even}, \\ -\dfrac{\Gamma\left(\frac{-1-\alpha}{2}-\left(\frac{k}{2}-l\right)\right)}{\Gamma\left(\frac{3+\alpha}{2}-\left(\frac{k}{2}-l\right)\right)}, & k \text{ odd}. \end{cases}$$

Hence, we obtain an equivalent but more convenient expression of $I_2$, containing absolute values as in (2.29):

$$I_2 = \begin{cases} \dfrac{\pi\alpha(1-\alpha)((-1+\alpha)k^2+4kl)\Gamma\left(\frac{-1-\alpha}{2}+\left|\frac{k}{2}-l\right|\right)}{2^{2+\alpha}\sin(\frac{\pi\alpha}{2})\Gamma(2-\alpha)\Gamma\left(\frac{3+\alpha}{2}+\left|\frac{k}{2}-l\right|\right)}, & k \text{ even}, \\ i\,\mathrm{sgn}(\frac{k}{2}-l)\dfrac{\pi\alpha(1-\alpha)((-1+\alpha)k^2+4kl)\Gamma\left(\frac{-1-\alpha}{2}+\left|\frac{k}{2}-l\right|\right)}{2^{2+\alpha}\cos(\frac{\pi\alpha}{2})\Gamma(2-\alpha)\Gamma\left(\frac{3+\alpha}{2}+\left|\frac{k}{2}-l\right|\right)}, & k \text{ odd}. \end{cases} \tag{2.30}$$

Putting (2.29) and (2.30) together,

$$d_{kl} = \begin{cases} \cot(\frac{\pi\alpha}{2})\alpha(1-\alpha)((1-\alpha)k^2-4kl) \\ \quad \cdot \dfrac{\Gamma\left(\frac{-1+\alpha}{2}+|l|\right)\Gamma\left(\frac{-1-\alpha}{2}+\left|\frac{k}{2}-l\right|\right)}{8\Gamma\left(\frac{3-\alpha}{2}+|l|\right)\Gamma\left(\frac{3+\alpha}{2}+\left|\frac{k}{2}-l\right|\right)}, & k \text{ even}, \\ i\,\mathrm{sgn}(\frac{k}{2}-l)\alpha(1-\alpha)((1-\alpha)k^2-4kl) \\ \quad \cdot \dfrac{\Gamma\left(\frac{-1+\alpha}{2}+|l|\right)\Gamma\left(\frac{-1-\alpha}{2}+\left|\frac{k}{2}-l\right|\right)}{8\Gamma\left(\frac{3-\alpha}{2}+|l|\right)\Gamma\left(\frac{3+\alpha}{2}+\left|\frac{k}{2}-l\right|\right)}, & k \text{ odd}. \end{cases} \tag{2.31}$$

Therefore, bearing in mind (2.23) and (2.24), we get (2.17), which concludes the proof of the theorem.

□

Remark: under the change of variable $x = \cot(s)$, the cosine-like and sine-like Higgins functions [7] are precisely $\cos(2ks)$ and $\sin((2k+2)s)$, which are eigenfunctions of the Hilbert transform [64]. Therefore, the first part of (2.18) follows also from the results in [64]. Note that we will study the fractional Laplacian of the Higgins functions in detail in Chapter 3.

### 2.1.5 Constructing an operational matrix

As explained above, in order to compute $(-\Delta)^{\alpha/2}u(x)$ for a given function $u(x)$, we first represent it as (2.14), then we apply Theorem 2.1.2 to each basic function $e^{iks}$. In this work, we have opted for a matrix approach, i.e., we have constructed a

differencing matrix $\mathbf{M}_\alpha \in \mathcal{M}_{(2N)\times(2N)}(\mathbb{C})$ based on Theorem 2.1.2, such that

$$\begin{pmatrix} (-\Delta)^{\alpha/2}u(s_0) \\ \vdots \\ (-\Delta)^{\alpha/2}u(s_{2N-1}) \end{pmatrix} \approx \mathbf{M}_\alpha \cdot \begin{pmatrix} \hat{u}(0) \\ \vdots \\ \hat{u}(N-1) \\ \hat{u}(-N) \\ \vdots \\ \hat{u}(-1) \end{pmatrix}, \tag{2.32}$$

where the nodes $s_j$ are defined in (2.1). It is vital to underline that, by choosing the appropriate strategy, the speed in the construction of $\mathbf{M}_\alpha$, and therefore, in the numerical computation of $(-\Delta)^{\alpha/2}u(x)$, can be increased by several orders of magnitude. Furthermore, that matrix needs to be computed just once, and then be reused whenever needed.

In order to generate $\mathbf{M}_\alpha$, we compute $(-\Delta)^{\alpha/2}(e^{iks})$ according to Theorem 2.1.2, for $k \in \{-N, \dots, N-1\}$. However, from (2.12), $(-\Delta)^{\alpha/2}(e^{i0s}) = (-\Delta)^{\alpha/2}(1) = 0$, and

$$\overline{(-\Delta)^{\alpha/2}(e^{iks})} = (-\Delta)^{\alpha/2}(e^{-iks}),$$

so we only need to calculate the cases with $k > 0$. Remark that the $N+1$th column could be assigned either $(-\Delta)^{\alpha/2}(e^{-iNs})$ or $(-\Delta)^{\alpha/2}(e^{iNs})$; in order to avoid choosing between one or the other, we fill it with zeros. Therefore, in order to create $\mathbf{M}_\alpha$, we only need to consider $k \in \{1, \dots, N-1\}$

Note that the implementation of Theorem 2.1.2 offers two difficulties: the need to evaluate the gamma function a very large number of times when $\alpha \neq 0$, and the fact that $l$ is taken all over $\mathbb{Z}$.

With respect to the gamma function, a fast and accurate implementation is usually available in every major scientific environment, such as MATLAB [8], which we use in this thesis. More precisely, in MATLAB, it is computed by the command `gamma`, which is based on algorithms outlined in [106]. However, using solely `gamma` to evaluate (2.17) is not numerically stable, because of the quick growth of `gamma` (for instance, `gamma(172)` yields infinity); therefore, even for rather small values of $l$, we get spurious `NaN` results, because we are dividing infinity by infinity. One possible solution would be to use the command MATLAB `gammaln`, which computes the natural logarithm of the gamma function, $\ln\Gamma$, i.e., the so-called log-gamma function:

$$\frac{\Gamma\left(\frac{-1+\alpha}{2}+|l|\right)\Gamma\left(\frac{-1-\alpha}{2}+\left|\frac{k}{2}-l\right|\right)}{\Gamma\left(\frac{3-\alpha}{2}+|l|\right)\Gamma\left(\frac{3+\alpha}{2}+\left|\frac{k}{2}-l\right|\right)} \equiv \exp\left[\ln\Gamma\left(\frac{-1+\alpha}{2}+|l|\right)\right.$$
$$\left. + \ln\Gamma\left(\frac{-1-\alpha}{2}+\left|\frac{k}{2}-l\right|\right) - \ln\Gamma\left(\frac{3-\alpha}{2}+|l|\right) - \ln\Gamma\left(\frac{3+\alpha}{2}+\left|\frac{k}{2}-l\right|\right)\right];$$

bear in mind that `gammaln` is not defined for negative values, so minor rewriting would be necessary in a few cases. However, in general, a much more convenient

solution is to use the basic property $\Gamma(z+1) = z\Gamma(z)$:

$$\begin{aligned}
\frac{\Gamma\left(\frac{-1+\alpha}{2}+|l|\right)}{\Gamma\left(\frac{3-\alpha}{2}+|l|\right)} &\equiv \frac{\frac{-3+\alpha}{2}+|l|}{\frac{1-\alpha}{2}+|l|}\cdot\frac{\Gamma\left(\frac{-3+\alpha}{2}+|l|\right)}{\Gamma\left(\frac{1-\alpha}{2}+|l|\right)},\\
\frac{\Gamma\left(\frac{-1-\alpha}{2}+\left|\frac{k}{2}-l\right|\right)}{\Gamma\left(\frac{3+\alpha}{2}+\left|\frac{k}{2}-l\right|\right)} &\equiv \frac{\frac{-3-\alpha}{2}+\left|\frac{k}{2}-l\right|}{\frac{1+\alpha}{2}+\left|\frac{k}{2}-l\right|}\cdot\frac{\Gamma\left(\frac{-3-\alpha}{2}+\left|\frac{k}{2}-l\right|\right)}{\Gamma\left(\frac{1+\alpha}{2}+\left|\frac{k}{2}-l\right|\right)},
\end{aligned} \tag{2.33}$$

where we consider separately the expressions containing $|l|$, and those containing $|k/2-l|$, because, for $|l| \gg 1$,

$$\frac{\frac{-3+\alpha}{2}+|l|}{\frac{1-\alpha}{2}+|l|} \approx 1, \qquad \frac{\frac{-3-\alpha}{2}+\left|\frac{k}{2}-l\right|}{\frac{1+\alpha}{2}+\left|\frac{k}{2}-l\right|} \approx 1,$$

so the factorizations in (2.33) are extremely stable from a numerical point of view. We apply recursively (2.33), until $|l| = 0$, and $|k/2-l| = 0$ (if $k$ even) or $|k/2-l| = 1/2$ (if $k$ odd). Therefore, for any $l$ and $k$, the evaluations of $\Gamma$ needed to compute the left-hand sides of (2.33) are just those in the quotients $\Gamma((-1+\alpha)/2)/\Gamma((3-\alpha)/2)$, $\Gamma((-1-\alpha)/2)/\Gamma((3+\alpha)/2)$ (if $k$ even), and $\Gamma(-\alpha/2)/\Gamma(2+\alpha/2)$ (if $k$ odd). Hence, taking into account that $\Gamma$ appears also in the definition of $c_\alpha$, it follows that the global number of evaluations of $\Gamma$ needed to compute (2.17) is very small, although, unfortunately, it does not seem possible to remove completely all the evaluations of $\Gamma$.

Bearing in mind the previous arguments, in order to approximate (2.17), we precompute recursively the right-hand sides of (2.33) for a large enough number of values, then store them in their respective vectors:

$$\begin{aligned}
\frac{\Gamma\left(\frac{-1+\alpha}{2}+|l|\right)}{\Gamma\left(\frac{3-\alpha}{2}+|l|\right)} &\equiv \frac{\Gamma\left(\frac{-1+\alpha}{2}\right)}{\Gamma\left(\frac{3-\alpha}{2}\right)}\prod_{m=0}^{|l|-1}\frac{\frac{-1+\alpha}{2}+m}{\frac{3-\alpha}{2}+m}, \quad \forall |l| \in \mathbb{N},\\
\frac{\Gamma\left(\frac{-1-\alpha}{2}+|\tilde{l}|\right)}{\Gamma\left(\frac{3+\alpha}{2}+|\tilde{l}|\right)} &\equiv \frac{\Gamma\left(\frac{-1-\alpha}{2}\right)}{\Gamma\left(\frac{3+\alpha}{2}\right)}\prod_{m=0}^{|\tilde{l}|-1}\frac{\frac{-1-\alpha}{2}+m}{\frac{3+\alpha}{2}+m}, \quad \forall |\tilde{l}| \in \mathbb{N},\\
\frac{\Gamma\left(\frac{-\alpha}{2}+|\tilde{l}|\right)}{\Gamma\left(2+\frac{\alpha}{2}+|\tilde{l}|\right)} &\equiv \frac{\Gamma\left(\frac{-\alpha}{2}\right)}{\Gamma\left(2+\frac{\alpha}{2}\right)}\prod_{m=0}^{|\tilde{l}|-1}\frac{\frac{-\alpha}{2}+m}{2+\frac{\alpha}{2}+m}, \quad \forall |\tilde{l}| \in \mathbb{N},
\end{aligned} \tag{2.34}$$

where the second expression is used in the cases with $k$ even, and the third one, in the cases with $k$ odd. Remark that the usage of absolute values makes trivial the computational evaluation of the vectors thus generated, because $|l|+1$, $|k/2-l|+1$ (if $k$ even), and $|k/2-l|+1/2$ (if $k$ odd) are precisely their respective indices.

With respect to $l$ spanning $\mathbb{Z}$, we decompose it as $l = l_1 N + l_2$, with $l_1 \in \mathbb{Z}$, and $l_2 \in \{-N/2, \ldots, N/2-1\}$ (if $N$ even), or $l_2 \in \{-(N-1)/2, \ldots, (N-1)/2\}$ (if $N$

odd), i.e., $l_2 \in \{-\lfloor N/2 \rfloor, \ldots, \lceil N/2 \rceil - 1\}$, considering both cases together. Note that we take $l_2$ between $-\lfloor N/2 \rfloor$ and $\lceil N/2 \rceil - 1$, rather than between 0 and $N-1$, because the numerical results appear to be slightly more accurate in that way. Then, we observe that

$$e^{i2ls_j} = e^{i2(l_1N+l_2)\pi(2j+1)/(2N)} = (-1)^{l_1} e^{i2l_2s_j},$$

i.e., aliasing occurs when evaluating $e^{i2ls}$ in the actual nodes. Therefore, we truncate $l_1$, i.e., take $l_1 \in \{-l_{lim}, \ldots, l_{lim}\}$, for $l_{lim}$ a large nonnegative integer. Then, (2.17) becomes

$$(-\Delta)^{\alpha/2}(e^{iks_j}) \approx \begin{cases} \dfrac{c_\alpha |\sin(s_j)|^{\alpha-1}}{8L^\alpha \tan(\frac{\pi\alpha}{2})} \displaystyle\sum_{l_2=-\lfloor N/2 \rfloor}^{\lceil N/2 \rceil - 1} \left[ \sum_{l_1=-l_{lim}}^{l_{lim}} a_{k,l_1,l_2} \right] e^{i2l_2s_j}, & k \text{ even}, \\ i\dfrac{c_\alpha |\sin(s_j)|^{\alpha-1}}{8L^\alpha} \displaystyle\sum_{l_2=-\lfloor N/2 \rfloor}^{\lceil N/2 \rceil - 1} \left[ \sum_{l_1=-l_{lim}}^{l_{lim}} a_{k,l_1,l_2} \right] e^{i2l_2s_j}, & k \text{ odd}, \end{cases} \tag{2.35}$$

where

$$a_{k,l_1,l_2} = \begin{cases} (-1)^{l_1}((1-\alpha)k^2 - 4k(l_1N+l_2)) \\ \cdot \dfrac{\Gamma\left(\frac{-1+\alpha}{2} + |l_1N+l_2|\right)\Gamma\left(\frac{-1-\alpha}{2} + \left|\frac{k}{2} - l_1N - l_2\right|\right)}{\Gamma\left(\frac{3-\alpha}{2} + |l_1N+l_2|\right)\Gamma\left(\frac{3+\alpha}{2} + \left|\frac{k}{2} - l_1N - l_2\right|\right)}, & k \text{ even}, \\ (-1)^{l_1}((1-\alpha)k^2 - 4k(l_1N+l_2))\operatorname{sgn}(\frac{k}{2} - l_1N - l_2) \\ \cdot \dfrac{\Gamma\left(\frac{-1+\alpha}{2} + |l_1N+l_2|\right)\Gamma\left(\frac{-1-\alpha}{2} + \left|\frac{k}{2} - l_1N - l_2\right|\right)}{\Gamma\left(\frac{3-\alpha}{2} + |l_1N+l_2|\right)\Gamma\left(\frac{3+\alpha}{2} + \left|\frac{k}{2} - l_1N - l_2\right|\right)}, & k \text{ odd}. \end{cases}$$

In this way, since we have used (2.34) to precompute the appearances of $\Gamma$ and have stored them in three vectors, the computation of

$$\sum_{l_1=-l_{lim}}^{l_{lim}} a_{k,l_1,l_2}$$

is reduced to sums and products and can be done in a very efficient way. Remark that, from the decomposition $l = l_1N + l_2$, it follows that, in order to generate the whole matrix $\mathbf{M}_\alpha$, the minimum length of the vectors generated from (2.34) is respectively $l_{lim}N + \lceil N/2 \rceil + 1$, $l_{lim}N + 2\lceil N/2 \rceil$ and $l_{lim}N + N$.

Finally, we perform the sum over $l_2$ in (2.35). Since

$$e^{i2l_2s_{N-1-j}} = e^{-i2l_2s_j}, \qquad e^{i2l_2s_{j+N}} = e^{i2l_2s_j},$$

it is enough to compute (2.35), for with $j \in \{0, \ldots, \lceil N/2 \rceil - 1\}$, and extend the results until $j = 2N-1$, by means of those symmetries. Alternatively, it is possible

to use the FFT, too.

Let us finish this section by mentioning that the case $\alpha = 1$ in (2.18) presents no difficulty. When $k$ is even, it is trivial to implement, and when $k$ is odd, we factorize and truncate $l$, as when $\alpha \neq 1$, obtaining

$$(-\Delta)^{1/2}(e^{iks_j}) \approx \begin{cases} \dfrac{|k|\sin^2(s_j)}{L} e^{iks_j}, & k \text{ even}, \\ \dfrac{ik}{L\pi}\left(\dfrac{-2}{k^2-4} - \displaystyle\sum_{l_2=-\lfloor N/2\rfloor}^{\lceil N/2\rceil-1}\left[\sum_{l_1=-l_{lim}}^{l_{lim}} b_{k,l_1,l_2}\right] e^{i2l_2s_j}\right), & k \text{ odd}, \end{cases} \tag{2.36}$$

with

$$b_{k,l_1,l_2} = \frac{4(-1)^{l_1}\operatorname{sgn}(l_1N+l_2)}{(k-2(l_1N+l_2))((k-2(l_1N+l_2))^2-4)}.$$

### 2.1.6 Some remarks on the convergence of the method

Assuming that the errors due to the floating point representation, evaluation of the involved functions, etc., are negligible, there are only two sources of error in the proposed method. On the one hand, we have the error in the numerical approximation of $(-\Delta)^{\alpha/2}(e^{iks})$ as defined in (2.19), when $\alpha = 1$, and (2.23), when $\alpha \neq 1$; and, on the other hand, the error due to the expansion of $u(s)$ as a Fourier series. With respect to the former, the error comes exclusively from truncating $l$ in (2.21), when $\alpha = 1$ and $k$ is odd (recall that the case with $\alpha = 1$ and $k$ even has a simple, exact form) or (2.24), when $\alpha \neq 1$.

The convergence of the case with $\alpha = 1$, $k$ odd, is simple to establish. Indeed, for a fixed value of $k$, from (2.21) and (2.22), taking $l_a, l_b \in \mathbb{N}$,

$$\begin{aligned} \left|\sin(s)\int_0^\pi \frac{\sin(\eta)e^{ik\eta}}{\sin(s-\eta)}d\eta - \sum_{l=-l_a}^{l_b} c_{kl}e^{i2ls}\right| &= \left|\sum_{l=-\infty}^{-l_a-1} c_{kl}e^{i2ls} + \sum_{l=l_b+1}^{\infty} c_{kl}e^{i2ls}\right| \\ &\leq \sum_{l=-\infty}^{-l_a-1}|c_{kl}| + \sum_{l=l_b+1}^{\infty}|c_{kl}| \\ &= \sum_{l=-\infty}^{-l_a-1}\frac{4}{|k-2l|\,|(k-2l)^2-4|} + \sum_{l=l_b+1}^{\infty}\frac{4}{|k-2l|\,|(k-2l)^2-4|}. \end{aligned} \tag{2.37}$$

Note that these bounds are not intended to be sharp, but they are enough to justify convergence (this is also valid for the case $\alpha \neq 0$ below). Now, remark that, when $\beta > 1$, for $N = 2, 3, \ldots$,

$$\frac{1}{(\beta-1)N^{\beta-1}} = \int_N^\infty \frac{dx}{x^\beta} \leq \sum_{n=N}^\infty \frac{1}{n^\beta} \leq \int_{N-1}^\infty \frac{dx}{x^\beta} = \frac{1}{(\beta-1)(N-1)^{\beta-1}},$$

where the sum is precisely the upper Riemann sum at the points $x = N, N+1, \ldots$ of the integral on the left-hand side, and the lower Riemann sum at the points $x = N-1, N, \ldots$ of the integral on the right-hand side. Therefore,

$$\int_N^\infty \frac{dx}{x^\beta} = \mathcal{O}\left(\frac{1}{N^{\beta-1}}\right), \tag{2.38}$$

and, from (2.37), we conclude that

$$\left|\sin(s)\int_0^\pi \frac{\sin(\eta)e^{ik\eta}}{\sin(s-\eta)}d\eta - \sum_{l=-l_a}^{l_b} c_{kl}e^{i2ls}\right| = \mathcal{O}\left(\frac{1}{l_a^2}\right) + \mathcal{O}\left(\frac{1}{l_b^2}\right),$$

which guarantees the convergence of (2.36), as $l_{lim}$ tends to infinity.

Regarding the case with $\alpha \neq 1$, for a fixed value of $k$, from (2.24) and (2.31), reasoning as in the case with $\alpha = 1$, we have

$$\begin{aligned}\left|\int_0^\pi \frac{\sin^\alpha(\eta)(-k^2\sin(\eta)+2ik\cos(\eta))e^{ik\eta}}{|\sin(s-\eta)|^{\alpha-1}}d\eta - \sum_{l=-l_a}^{l_b} d_{kl}e^{i2ls}\right| \\ \leq \sum_{l=-\infty}^{-l_a-1}|d_{kl}| + \sum_{l=l_b+1}^{\infty}|d_{kl}|.\end{aligned} \tag{2.39}$$

In order to obtain how $d_{kl}$ decays with $l$, we use Stirling's formula for the gamma function [107]:

$$\Gamma(x) \sim x^{x-1}e^{-x}\sqrt{2\pi x}, \quad x \gg 1,$$

to substitute the occurrences of the gamma function in (2.31) with their asymptotic equivalents:

$$\begin{aligned}&l\frac{\Gamma\left(\frac{-1+\alpha}{2}+|l|\right)\Gamma\left(\frac{-1-\alpha}{2}+\left|\frac{k}{2}-l\right|\right)}{\Gamma\left(\frac{3-\alpha}{2}+|l|\right)\Gamma\left(\frac{3+\alpha}{2}+\left|\frac{k}{2}-l\right|\right)} \\ &\quad\sim l\frac{\left(\frac{-1+\alpha}{2}+|l|\right)^{\frac{-3+\alpha}{2}+|l|}e^{\frac{1-\alpha}{2}-|l|}\sqrt{2\pi\left(\frac{-1+\alpha}{2}+|l|\right)}}{\left(\frac{3-\alpha}{2}+|l|\right)^{\frac{1-\alpha}{2}+|l|}e^{\frac{-3+\alpha}{2}-|l|}\sqrt{2\pi\left(\frac{3-\alpha}{2}+|l|\right)}} \\ &\qquad\cdot\frac{\left(\frac{-1-\alpha}{2}+\left|\frac{k}{2}-l\right|\right)^{\frac{-3-\alpha}{2}+\left|\frac{k}{2}-l\right|}e^{\frac{1+\alpha}{2}-\left|\frac{k}{2}-l\right|}\sqrt{2\pi\left(\frac{-1-\alpha}{2}+\left|\frac{k}{2}-l\right|\right)}}{\left(\frac{3+\alpha}{2}+\left|\frac{k}{2}-l\right|\right)^{\frac{1+\alpha}{2}+\left|\frac{k}{2}-l\right|}e^{\frac{-3-\alpha}{2}-\left|\frac{k}{2}-l\right|}\sqrt{2\pi\left(\frac{3+\alpha}{2}+\left|\frac{k}{2}-l\right|\right)}}.\end{aligned}$$

Therefore, when $|l| \gg 1$,

$$l\frac{\Gamma\left(\frac{-1+\alpha}{2}+|l|\right)\Gamma\left(\frac{-1-\alpha}{2}+\left|\frac{k}{2}-l\right|\right)}{\Gamma\left(\frac{3-\alpha}{2}+|l|\right)\Gamma\left(\frac{3+\alpha}{2}+\left|\frac{k}{2}-l\right|\right)}=\mathcal{O}\left(\frac{1}{l^3}\right),$$

and (2.39) becomes

$$\left|\int_0^\pi \frac{\sin^\alpha(\eta)(-k^2\sin(\eta)+2ik\cos(\eta))e^{ik\eta}}{|\sin(s-\eta)|^{\alpha-1}}d\eta-\sum_{l=-l_a}^{l_b} d_{kl}e^{i2ls}\right|=\mathcal{O}\left(\frac{1}{l_a^2}\right)+\mathcal{O}\left(\frac{1}{l_b^2}\right),$$

which guarantees the convergence of (2.35), as $l_{lim}$ tends to infinity.

In what respects the expansion of a given function $u(s)$ as a Fourier series, recall that, under the change of variable $x = L\cot(s)$, $u(x)$ becomes $u(s)$, with $s \in [0,\pi]$, and we have to extend it to $s \in [0, 2\pi]$. Even if there are infinitely many ways to extend a function defined over half a period to the whole period, we comment here only on the most common option, which is an even extension at $s = \pi$. Such extension gives rise to a series of cosines, which is equivalent to expanding $u(x)$ as a series of rational Chebyshev functions $TB_k(x)$ defined in (2.8). As mentioned above, this family of functions is very adequate to represent regular functions having different types of decay at infinity [63], or, in other words, we may write

$$u(x) \equiv \sum_{k=0}^{\infty} a_k TB_k(x), \quad \forall x \in \mathbb{R}, \tag{2.40}$$

with the coefficients $a_k$ decaying fast enough for a variety of functions; we refer the reader to [2], and the references therein (especially, [2, Table 17.4]), for a detailed account on the applicability of $\{TB_k(x)\}$. Furthermore, the fact that the right-hand side of (2.40) converges to $u(x)$ implies automatically that the method proposed in this chapter converges to $(-\Delta)^{\alpha/2}u(x)$, too.

In the following section, we consider a few examples that clarify several aspects related to convergence.

## 2.2 Numerical tests

In this section, we take $x_c = 0$ in (2.13), because the test functions are symmetrical with respect to the origin. We have first considered two functions with polynomial decay:

$$u_1(x) = \frac{x^2-1}{x^2+1}, \qquad u_2(x) = \frac{2x}{x^2+1},$$

where the first one tends to 1 as $\mathcal{O}(1/x^2)$, whereas the second one tends to 0 as $\mathcal{O}(1/x)$. The choice of these two functions has also been motivated by the fact that, under the change of variable $x = \cot(s)$, with $L = 1$, they become respectively

$u_1(s) = \cos(2s)$ and $u_2(s) = \sin(2s)$, i.e., the real and imaginary parts of $e^{i2s}$, so we have computed the fractional Laplacian of $u_1(x) + iu_2(x)$. Using MATHEMATICA applied to (2.6), and further simplifying the result by hand, we get

$$(-\Delta)^{\alpha/2}(u_1(x) + iu_2(x)) = -\frac{2\Gamma(1+\alpha)}{(1+ix)^{1+\alpha}},$$

or, in the $s$ variable,

$$(-\Delta)^{\alpha/2}(e^{i2s}) = -\frac{2\Gamma(1+\alpha)}{(1+i\cot(s))^{1+\alpha}} = -2\Gamma(1+\alpha)(-i\sin(s)e^{is})^{1+\alpha}; \tag{2.41}$$

note that, when $\alpha = 1$, we recover (2.18). In general, as we will see in Chapter 3, it is possible to compute explicitly $(-\Delta)^{\alpha/2}(e^{iks})$ when $k$ is even (and, indeed, we will compute $(-\Delta)^{\alpha/2}(e^{i2s})$ also using analytic techniques), although the complexity of the resulting expressions quickly grows with $k$. On the other hand, we have been unable to obtain a compact formula for $(-\Delta)^{\alpha/2}(e^{iks})$, when $k$ is odd, either by hand or using symbolic computation.

| $N$ | $l_{lim}$ | $l_{total}$ | Error | $l_{lim}$ | $l_{total}$ | Error |
|---|---|---|---|---|---|---|
| 4 | 300 | 4244 | $4.8893 \cdot 10^{-12}$ | 530 | 4244 | $5.0268 \cdot 10^{-13}$ |
| 8 | 240 | 6888 | $4.9423 \cdot 10^{-12}$ | 430 | 6888 | $4.8097 \cdot 10^{-13}$ |
| 16 | 200 | 11536 | $4.8413 \cdot 10^{-12}$ | 360 | 11536 | $4.9461 \cdot 10^{-13}$ |
| 32 | 170 | 19232 | $4.5606 \cdot 10^{-12}$ | 300 | 19232 | $5.0138 \cdot 10^{-13}$ |
| 64 | 140 | 32064 | $4.9236 \cdot 10^{-12}$ | 250 | 32064 | $5.0184 \cdot 10^{-13}$ |
| 128 | 120 | 53888 | $4.5427 \cdot 10^{-12}$ | 210 | 53888 | $5.0219 \cdot 10^{-13}$ |
| 256 | 100 | 92416 | $4.6956 \cdot 10^{-12}$ | 180 | 92416 | $5.0570 \cdot 10^{-13}$ |
| 512 | 80 | 154112 | $5.7013 \cdot 10^{-12}$ | 150 | 154112 | $5.0823 \cdot 10^{-13}$ |
| 1024 | 70 | 287744 | $4.8721 \cdot 10^{-12}$ | 140 | 287744 | $5.2887 \cdot 10^{-13}$ |

TABLE 2.1: Maximum global error, given by (2.42), between the numerical approximation of $(-\Delta)^{\alpha/2}(e^{i2s})$, given by (2.35), and its exact value, given by (2.41). We have considered different values of $N$ and, for each $N$, a couple of values of $l_{lim}$. For comparison purposes, we also offer the total number of values of $l$ considered, $l_{total} \equiv (2l_{lim}+1)N$.

Taking different values of $N$ and $l_{lim}$, we have approximated $(-\Delta)^{\alpha/2}(e^{i2s})$ numerically, which we denote as $[(-\Delta)^{\alpha/2}]_{num}(e^{i2s})$, by means of (2.35), without generating the whole matrix $\mathbf{M}_\alpha$, for $\alpha \in \{0.01, 0.02, \ldots, 1.99\}$, except for the case $\alpha = 1$, which is trivial (altogether, 1998 different values of $\alpha$). Then, we have compared the results with their exact value of $[(-\Delta)^{\alpha/2}]_{exact}(e^{i2s})$ given by (2.41), and computed the discrete $L^\infty$-norm of the error as a function of $\alpha$. In Table 2.1, we show the maximum global value of the norm considering all $\alpha$, i.e.,

$$\begin{aligned}\max_\alpha \| &[(-\Delta)^{\alpha/2}]_{num}(e^{i2s}) - [(-\Delta)^{\alpha/2}]_{exact}(e^{i2s})\| \\ &= \max_\alpha \max_j \left| [(-\Delta)^{\alpha/2}]_{num}(e^{i2s_j}) - [(-\Delta)^{\alpha/2}]_{exact}(e^{i2s_j}) \right|,\end{aligned} \tag{2.42}$$

for different values of $N$ and $l_{lim}$. For comparison, we also offer $l_{total} \equiv (2l_{lim}+1)N$, which is the exact number of values of $l$ taken, i.e., $l \in \{-l_{total}/2, \ldots, l_{total}/2-1\}$. The results reveal that the value of $l_{lim}$ necessary to achieve an error of the order of $5 \cdot 10^{-13}$ slowly decreases as $N$ is doubled, but, more importantly, the accuracy of the method does not deteriorate, as $N$ increases.

In order to better understand the choice of $l_{lim}$ on the accuracy of the results, we have approximated $(-\Delta)^{\alpha/2}(e^{i2s})$, for $l_{lim} \in \{0, 1, \ldots, 1000\}$, and have plotted in Figure 2.1 the corresponding maximum global error given by (2.42). As we can see, the errors quickly decay from $l_{lim} = 0$, with an error of $3.1960 \cdot 10^{-3}$, to $l_{lim} = 210$, with an error of $5.0219 \cdot 10^{-13}$, from which it remains constant up to infinitesimal variations. This is important, because it shows that (2.35) is numerically stable, even for larger values of $l_{lim}$. A practical consequence of this is that, in case of doubt, it is safe to take a rather large value of $l_{lim}$.

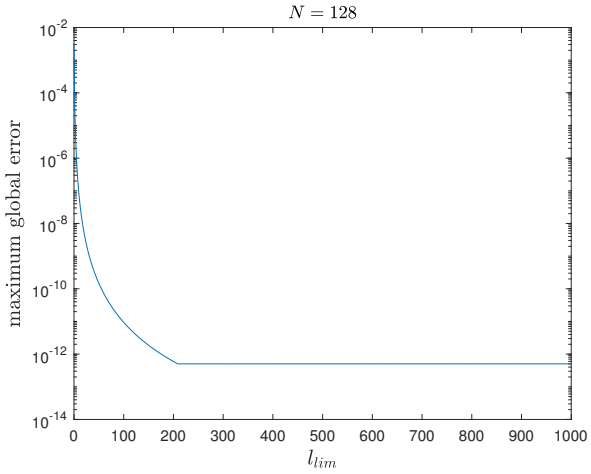


FIGURE 2.1: Maximum global error for $N = 128$, as a function of $l_{lim}$.

Let us consider now a function with Gaussian decay,

$$u_3 = \exp(-x^2),$$

such that (see, for instance, [11, pp. 29-30])

$$(-\Delta)^{\alpha/2} u_3(x) = \frac{2^\alpha \Gamma(1/2+\alpha/2)}{\sqrt{\pi}} {}_1F_1(1/2+\alpha/2,\ 1/2,\ -x^2), \tag{2.43}$$

where ${}_1F_1$ is the Kummer confluent hypergeometric function (see for instance [4, Ch. 13]), which can be evaluated, among others, by MATLAB (with the command `hypergeom`) and MATHEMATICA (with the command `Hypergeometric1F1`), even if its evaluation is extremely time-consuming.

In this example, we have generated the whole matrix $\mathbf{M}_\alpha$ applied to the Fourier expansion (2.14) of $u_3(s)$, as in (2.32). Remember that, since $s \in [0, \pi]$, we have to extend it to $s \in [0, 2\pi]$. In general, the most common option is an even extension at $s = \pi$, which yields a function that is at least continuous in $s \in [0, 2\pi]$, and

can be represented as a cosine series in $s$. However, in some cases, there are extensions that are smoother than the even one (see v.g. [108] and [109]), causing the Fourier coefficients in (2.14) to decay faster. This is not a minor point, because, even if $(-\Delta)^{\alpha/2}(e^{iks})$ can be computed accurately, as we have proved in Section 2.1.6 and confirmed in the previous example, the overall quality of the results depends also on the adequacy of the representation (2.14).

| $N$ | Error (even) | Error (odd) |
|---|---|---|
| 4 | $3.8426 \cdot 10^{-1}$ | $4.8492 \cdot 10^{-1}$ |
| 8 | $1.1222 \cdot 10^{-1}$ | $1.3210 \cdot 10^{-1}$ |
| 16 | $1.4269 \cdot 10^{-2}$ | $1.7825 \cdot 10^{-2}$ |
| 32 | $4.0393 \cdot 10^{-4}$ | $4.7926 \cdot 10^{-4}$ |
| 64 | $1.4351 \cdot 10^{-6}$ | $1.6891 \cdot 10^{-6}$ |
| 128 | $1.5947 \cdot 10^{-10}$ | $1.8755 \cdot 10^{-10}$ |
| 256 | $8.3982 \cdot 10^{-12}$ | $2.5453 \cdot 10^{-11}$ |

TABLE 2.2: Maximum global error, between the numerical approximation of $(-\Delta)^{\alpha/2}(e^{-x^2})$, and its exact value, given by (2.43), for different values of $N$, considering an even extension and an odd extension. $l_{lim} = 500$.

In this example, since $u_3(x)$ tends to zero as $x \to \pm\infty$ (or $s \to 0^+$ and $s \to \pi^-$), we have considered both an even and an odd extension at $s = \pi$, i.e., such that $u_3(\pi^+) = u_3(\pi^-)$ and $u_3(\pi^+) = -u_3(\pi^-)$, respectively. For this function, in the even case, we also have that $u_3(s + \pi) = u_3(s)$, which implies that only even frequencies appear in (2.14); whereas in the odd case, $u_3(s + \pi) = -u_3(s)$, so only odd frequencies appear in (2.14). As a consequence, besides comparing two types of extensions, we are also testing the even and odd cases in (2.35) and (2.36).

We have approximated $(-\Delta)^{\alpha/2}u_3(x)$ for $\alpha \in \{0.01, 0.02, \ldots, 1.99\}$ (including the case $\alpha = 1$), for $L = 1$, $l_{lim} = 500$, and different values of $N$. In Table 2.2, we give the maximum global errors computed as in (2.42). As we can see, the even extension provides only slightly better results, and the errors quickly decays, as $N$ increase.

Even if we have taken so far $L = 1$, this is usually by no means the best option, as we can see in Figure 2.2, where we have plotted the maximum global error for $N = 64$, and $L \in \{0.1, 0.2, \ldots, 10\}$. The results for the even extension and the odd extension are again similar, and the best errors are achieved in both cases at $L = 4.6$, and are respectively $3.8400 \cdot 10^{-13}$ and $3.9466 \cdot 10^{-13}$. Therefore, a good choice of $L$ can improve drastically the accuracy of the results.

Although there are some theoretical results [110], the optimal value of $L$ depends on more than one factor: number of points, class of functions, type of problem, etc (see also [63, 100]). For instance, in the case of $(-\Delta)^{\alpha/2}$, the best choice of $L$ might depend on $\alpha$, too. However, a good working rule of thumb seems to be that the absolute value of a given function at the extreme grid points is smaller than a threshold. On the other hand, the Fourier representation (2.14), together with (2.13), makes straightforward to change $L$ (and $x_c$ or $N$), which can be convenient in evolution problems. Let us recall that, given a function $u(x)$, we are considering a

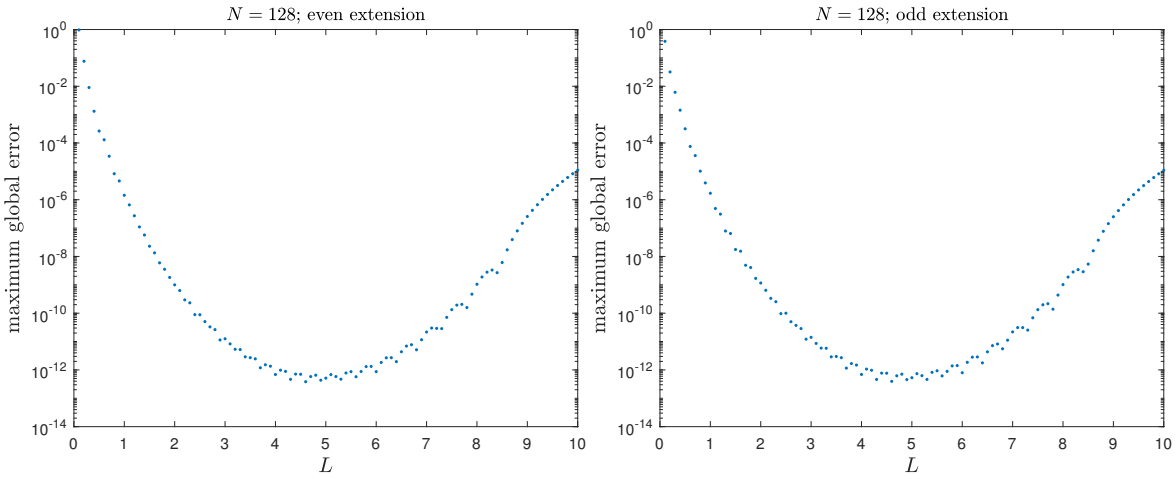


FIGURE 2.2: Maximum global error for $N = 64$, and different values of $L$: $L \in \{0.1, 0.2, \ldots, 10\}$, considering an even extension and an odd extension.

spectral interpolant such that

$$u(x) \approx \sum_{k=-N}^{N-1} \hat{u}(k) e^{ik \operatorname{arccot}((x-x_c)/L)}, \tag{2.44}$$

and, to determine $\{\hat{u}(k)\}$ we ask (2.44) to be an equality at the nodes $x_j = x_c + L\cot(s_j)$, which yields (2.16). Therefore, if we choose new values of $L$ and $x_c$, say $L_{new}$ and $x_{c,new}$, and want to approximate $u(x)$ at the corresponding nodes $x_{new,j} = x_{c,new} + L_{new}\cot(s_j)$ by using *spectral interpolation*, it is enough to evaluate the right-hand side of (2.44) at those nodes:

$$u(x_{new,j}) \approx \sum_{k=-N}^{N-1} \hat{u}(k) e^{ik \operatorname{arccot}((x_{c,new}-x_c+L_{new}\cot(s_j))/L)},$$

where, when $0 \leq j \leq N-1$, we consider the arccot function to be defined in $[0, \pi)$, and, when $N \leq j \leq 2N-1$, to be defined in $[\pi, 2\pi)$. Moreover, from $\{u(x_{new,j})\}$, we obtain the corresponding $\{\hat{u}_{new}(k)\}$ by using again a pseudospectral approach, i.e., by imposing that (2.44) with the updated $L_{new}$ and $x_{c,new}$ is an equality at $x = x_{j,new}$, for all $j$:

$$u(x_{new,j}) = \sum_{k=-N}^{N-1} \hat{u}_{new}(k) e^{ik \operatorname{arccot}((x_{new,j}-x_{c,new})/L_{new})},$$

so the coefficients $\hat{u}_{new}(k)$ are given by (2.16), introducing $u(x_{new,j})$ in the place of $u(s_j)$, and taking $L_{new}$ and $x_{c,new}$. Then,

$$u(x) \approx \sum_{k=-N}^{N-1} \hat{u}_{new}(k) e^{ik \operatorname{arccot}((x-x_{c,new})/L_{new})}.$$

Finally, (2.44) allows also changing $N$; e.g., if $N$ is increased, we just add some extra $\hat{u}(k)$ equal to zero; if it is decreased, we remove some $\hat{u}(k)$. In all the cases considered, it is important to choose the new values of $L$, $x_c$ and $N$, in such a way that there is no loss of accuracy.

## 2.3 A numerical test for the fractional Fisher's equation (1.11) with very slowly varying initial conditions

As an illustration of the method presented in Section 2.1, we will simulate numerically the one-dimensional nonlinear evolution equation (1.11) in the monostable case, i.e., with the following nonlinear source term:

$$f(u) = u(1-u). \tag{2.45}$$

The interest on this simulation lies on the fact that (1.11) exhibits accelerating fronts, and our method allows to capture this rapid propagation. Moreover, the acceleration can be computed and contrasted with theoretical results. Let us first recall some aspects of front propagation for this model with $\alpha = 2$.

The front propagation properties of the homogeneous local Fisher equation,

$$\frac{\partial u}{\partial t} = \frac{\partial^2 u}{\partial x^2} + f(u), \quad x \in \mathbb{R}, \tag{2.46}$$

have been widely studied. We recall that the steady state $u \equiv 0$ is linearly unstable, whereas $u \equiv 1$ is stable. An initial perturbation of $u \equiv 0$ evolves to a front that characterizes the transition from the unstable state $u = 0$ to the stable one $u = 1$; and the front-like solution invades the unstable state at a constant speed, as $t \to +\infty$ (see, e.g., [111]). For initial data that decay in $x$ as an exponential or faster, the front speed for the specific nonlinearity $f(u) = u(1-u)$ studied in [42] can be determined by linear arguments (see, e.g., [112, 113]). The two main formal approaches are reviewed in, e.g., [114].

One approach assumes that the solution behaves as a traveling wave solution as $t \to \infty$; hence, setting $u \sim U(z)$, $z = x - s(t)$, $s(t) \sim c\,t$, as $t \to +\infty$, gives the following traveling-wave ordinary differential equation:

$$\frac{d^2U}{dz^2} + c\frac{dU}{dz} + f(U) = 0, \tag{2.47}$$

with far-field conditions

$$U \to 1, \text{ as } z \to -\infty, \quad \text{and} \quad U \to 0, \text{ as } z \to +\infty. \tag{2.48}$$

The linearization analysis leads to the behaviors

$$U \sim A_{\pm} e^{-\frac{1}{2}\left(c \pm \sqrt{c^2-4}\right)z}, \quad c > 2, \tag{2.49}$$

$$U \sim (A + Bz)e^{-z}, \qquad c = 2, \tag{2.50}$$

as $z \to +\infty$, where $A_{\pm}$, $A$ and $B$ are constants. Here, $c \geq 2$ is necessary for non-negative wave fronts, and the wave speed selected for faster than exponentially decaying initial data is the minimal one. Rigorous arguments that show this heuristics are proved in, e.g., [115] (see also [113] for a generalization of the result to two dimensions).

For slower decaying initial conditions, speeds faster than $c = 2$ are realized. Specifically, for initial data of the form

$$u_0(x) \sim e^{-\lambda x}, \quad \text{as } x \to +\infty, \tag{2.51}$$

where $0 < \lambda < \lambda_c = 1$ (i.e., the initial condition decays more slowly than the far-field behavior (2.50)), the solution behaves like

$$v \sim e^{-\lambda(\zeta - c)t}, \quad \text{as } x \to +\infty, \tag{2.52}$$

with $c = \lambda + 1/\lambda$ (hence, $\lambda = (c - \sqrt{c^2 - 4})/2$, which corresponds to slow far-field decay in (2.49)).

Sufficient conditions on the initial data for the convergence to traveling-wave solutions with different waves speeds, as $t \to +\infty$, were established in [116, 117] (see also [118], for a slightly more general nonlinearity, and a different approach). The dynamic stability of traveling wave solutions has been studied by a number of authors, and we refer the reader to the seminal paper [5] (see also [119–121], and the references therein for details).

There has also been an effort in obtaining numerical schemes, in particular pseudospectral methods, that capture these fast traveling waves (associated to exponentially decaying initial conditions). Such solutions appear to be very steep, when a large reaction coefficient is considered (see, e.g., [122], for Fisher's equation, and [123], for Nagumo's equation).

Another important feature of (2.46) regarding solutions invading the unstable state (but which is a less studied phenomenon) is the existence of accelerating fronts (see, e.g., [118, 124, 125]). In particular, accelerating fronts ensue for initial conditions that decay to zero slower than exponentially. In this case, the long time behavior is given by the balance equation $u_t = u$; hence, as $t \to \infty$, $u(x,t) \sim e^t u(x,0)$ gives the dominant behavior (see [118, 124]). Then, for initial conditions of the form $u(x,0) \sim 1/x^{\sigma}$, as $x \to \infty$, this implies that the invasion into $u = 0$, happens with $x = O(e^{t/\sigma})$, as $t \to \infty$. These results are proved rigorously in [125].

We restrict ourselves to the example of slow decaying (according to [56]) initial conditions; more precisely, we consider

$$u(x,0) = \left(\frac{1}{2} - \frac{x}{2\sqrt{1+x^2}}\right)^{\alpha}.$$

In order to check that, for $\alpha \in (0,2)$, the propagation has indeed speed that increases exponentially with time, we track the evolution of $x_{0.5}(t)$, which denotes the value of $x$ such that $u(x,t) = 0.5$, and gives an approximation of the position of the front. To obtain $x_{0.5}$, we apply a bisection method: we find the value of $j$ for which $u(x_{j+1}) < 0.5 < u(x_j)$; then, we approximate $u((x_j + u_{j+1})/2)$ by spectral interpolation, etc., until convergence is achieved.

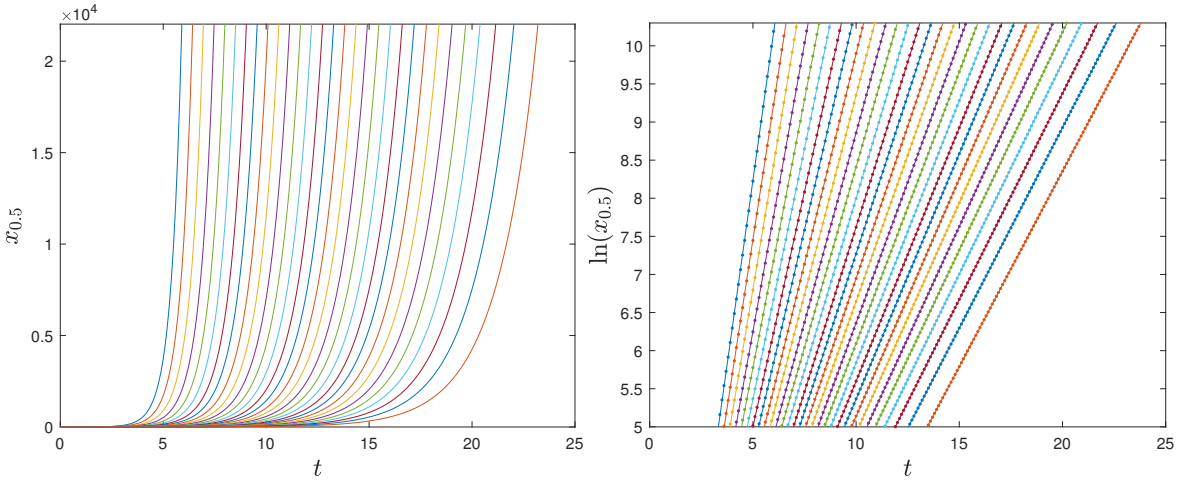


FIGURE 2.3: $\alpha = 0.5, 0.55, \ldots, 1.95$, $L = 10^3/\alpha^3$, $\Delta t = 0.01$ and $N = 1024$. Left: $x_{0.5}(t)$ against $t$. Right: $\ln(x_{0.5}(t))$ against $t$, and the corresponding least-square fitting lines. In both subfigures, the curves are ordered according to $\alpha$: the left-most ones correspond to $\alpha = 0.5$, and the right-most ones, to $\alpha = 1.95$.

In all the numerical experiments, we have considered an even extension at $s = \pi$, which is enough for our purposes, taken $l_{lim} = 500$ in (2.35), and used the classical fourth-order Runge-Kutta scheme (see, e.g., [6, p. 226]) to advance in time. We have done the numerical simulation for $\alpha = 0.5, 0.55, \ldots, 1.95$, taking $\Delta t = 0.01$ and $N = 1024$. Since the exponential behavior of $x_{0.5}(t)$ appears earlier for smaller $\alpha$, larger values of $L$ appear to be convenient in that case. In this example, after a couple of trials, we have found that taking $L = 1000/\alpha^3$ produces satisfactory results. On the left-hand side of Figure 2.3, we have plotted $x_{0.5}(t)$ against $t$. On the right-hand side of Figure 2.3, we have plotted $\ln(x_{0.5}(t))$ against $t$, omitting the initial times, so the exponential regime is clearly observable; in all cases, the points are separated by time increments of 0.1, and, for each value of $\alpha$, the accompanying line is precisely the least-square fitting line, which shows that the linear alignment is almost perfect.

In Figure 2.4, we have plotted with respect to $\alpha$ the slopes of the least-square fitting lines corresponding to the right-hand side of Figure 2.3, which we denote as $\sigma_{0.5}$;

observe that the colors of the stars are in agreement with their corresponding curves in Figure 2.3. We have also plotted the curve $1/\alpha$, using a dashed-dotted black line. The results show that the agreement of $\sigma_{0.5}$ with respect to $1/\alpha$ improves, as $\alpha \to 2^-$: on the one hand, when $\alpha = 0.5$, $\sigma_{0.5} = 1.9346$, and $1/0.5 = 2$; on the other hand, when $\alpha = 1.95$, $\sigma_{0.5} = 0.51277$, and $1/1.95 = 0.51282$. Therefore, the numerical experiments seem to suggest that

$$x_{0.5}(t) \sim e^{\sigma_{0.5} t} \sim e^{t/\alpha} \Longrightarrow c(t) \approx x'_{0.5}(t) \sim e^{t/\alpha},$$

which is in good agreement with [56], because, from (2.45), $f'(0) = 1$.

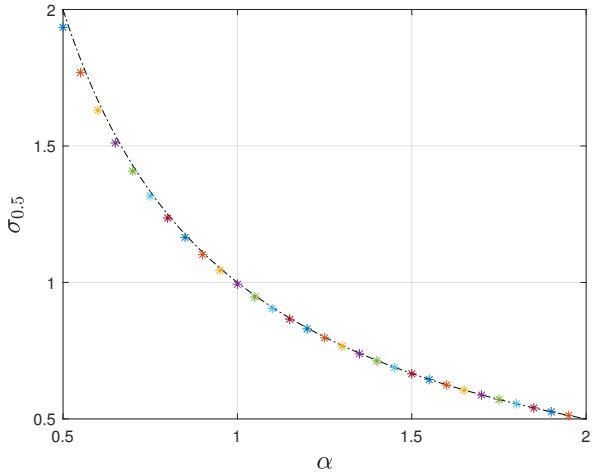


FIGURE 2.4: Slopes of the least-square fitting lines, as obtained in the right-hand side of Figure 2.3; the colors of the stars are in agreement with their corresponding curves in Figure 2.3. The dashed-dotted black curve is the plot of $1/\alpha$.

In order to see whether the results for $\alpha = 0.5$ can be improved, we have repeated the simulations for that case, taking $L = 10000$, $\Delta t = 0.01$, $N = 8192$. Even if, at first sight, these values could be deemed excessive, they are not, because we are able to reach $t = 9$, instant at which $x_{0.5}(9)$ is greater than $10^7$. Indeed, in order to capture accurately the exponential behavior, it is convenient to advance until times as large as possible. On the left-hand side of Figure 2.5, we have plotted $x_{0.5}(t)$, for $t \in [0, 9]$; on the right-hand side, $\ln(x_{0.5}(t))$, for $t \in [5, 9]$, obtaining again an almost perfect linear fitting. Furthermore, in this case, $\sigma_{0.5} = 1.9865$, which is remarkably closer to the predicted value $1/0.5 = 2$ than in Figure 2.4. Therefore, in order to approximate accurately $\sigma_{0.5}$ for values of $\alpha$ smaller than 0.5, it will be convenient to take even larger values of $N$ and $L$.

If we intend to know the type of decay that the wave front has, one option could be to choose those nodes in the abscissa whose images capture the asymptotic behavior toward the values 1 and 0, i.e., $u(x_i) \to 1$, as $x_i \to -\infty$, and $u(x_j) \to 0$, as $x_j \to \infty$. Here, the integers $i$ and $j$ refer, respectively, to the first $m$ and the last $n$ nodes taken into account to study such asymptotic behavior of $u$. With some abuse of notation, $u_i \equiv u(x_i)$ and $u_j \equiv u(x_j)$. If we consider $t > 1$, $x_i \ll 1$ and $x_j \gg 1$, it is possible to assume that the wave front $u_i$ behaves as $\zeta_1 {x_i}^{-\beta_1}$, as $x_i \to -\infty$, and $u_j$ behaves as

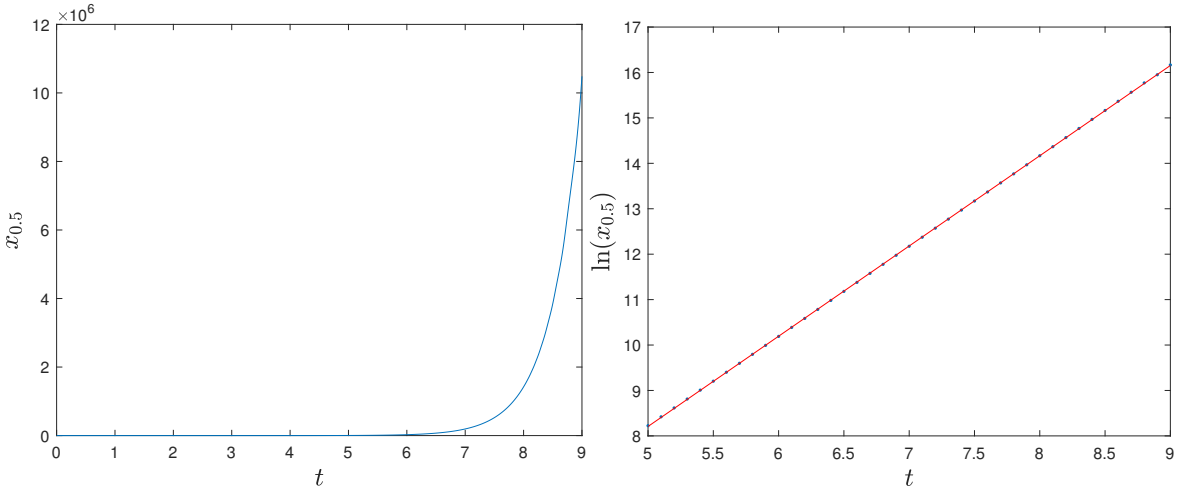


FIGURE 2.5: $\alpha = 0.5$, $L = 10^4$, $\Delta t = 5 \cdot 10^{-3}$, and $N = 8192$. Left: $x_{0.5}(t)$ against $t \in [0, 9]$. Right: $\ln(x_{0.5}(t))$ against $t \in [5, 9]$, and the corresponding least-square fitting line.

$\zeta_0 x_j^{-\beta_0}$, as $x_j \to \infty$. Therefore, for (1.11) with the source term (2.45), a least-square fitting can be carried out for both cases. On the left-hand side of Figure 2.6, we plot the fitting curve when $x_i \ll 1$, taking $m = 65$ nodes. On the right-hand side of Figure 2.6, we plot the fitting curve when $x_j \gg 1$, taking $n = 65$ nodes as well.

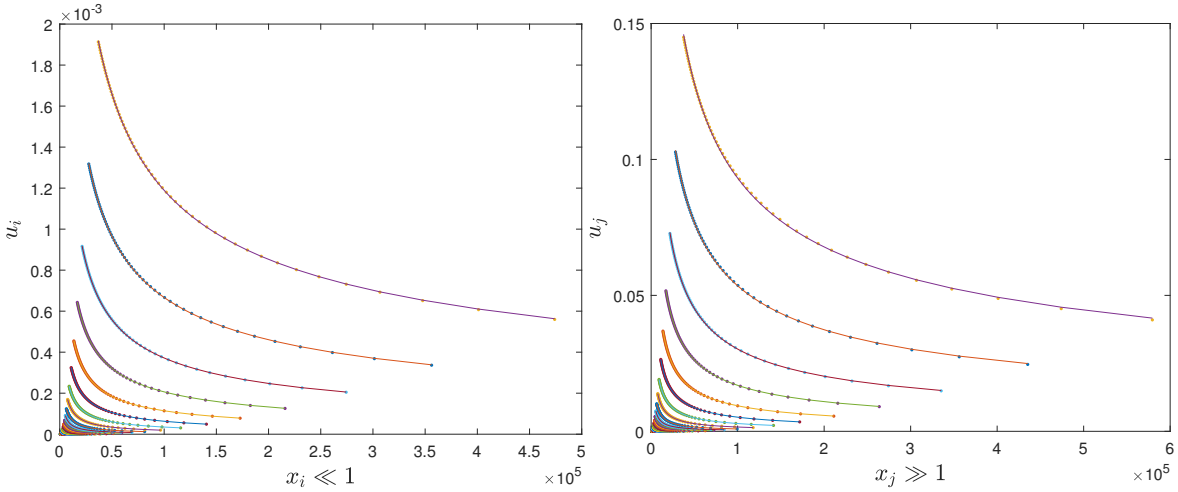


FIGURE 2.6: $\alpha = 0.5, 0.55, \ldots, 1.95$, $L = 10^3/\alpha^3$, $\Delta t = 0.01$ and $N = 1024$. Left: least-square fitting for $u_i \to 1$ against $x_i \ll 1$. Right: least-square fitting for $u_j \to 0$ against $x_j \gg 1$.

Each fitting (for each value of $\alpha$) yields a residual, which we measure through the Euclidean norm given by

$$\|r_i^\alpha\|_2 = \frac{1}{m}\sqrt{\sum_{i=1}^{m} |r_i^\alpha|^2}, \quad \text{when } x_i \ll 1,$$

and

$$\|r_j^\alpha\|_2 = \frac{1}{n}\sqrt{\sum_{j=1}^{n} |r_j^\alpha|^2}, \quad \text{when } x_j \gg 1,$$

where

$$r_i^\alpha = u_i - \zeta_1 {x_i}^{-\beta_1}$$

and

$$r_j^\alpha = u_j - \zeta_0 {x_j}^{-\beta_0},$$

respectively. On the left-hand side of Figure 2.7, we have plotted the norm of the residuals $\|r_i^\alpha\|_2$ with respect to $\alpha$, achieving a maximum value of $2.4 \cdot 10^{-7}$; and, on the right-hand side of the same figure, the norm of the residuals $\|r_j^\alpha\|_2$ with respect to $\alpha$, achieving a maximum value of $7.3 \cdot 10^{-5}$. Hence, these results suggest a potential decay for this slowly varying initial condition of the non-linear Fisher-KPP equation (2.6).

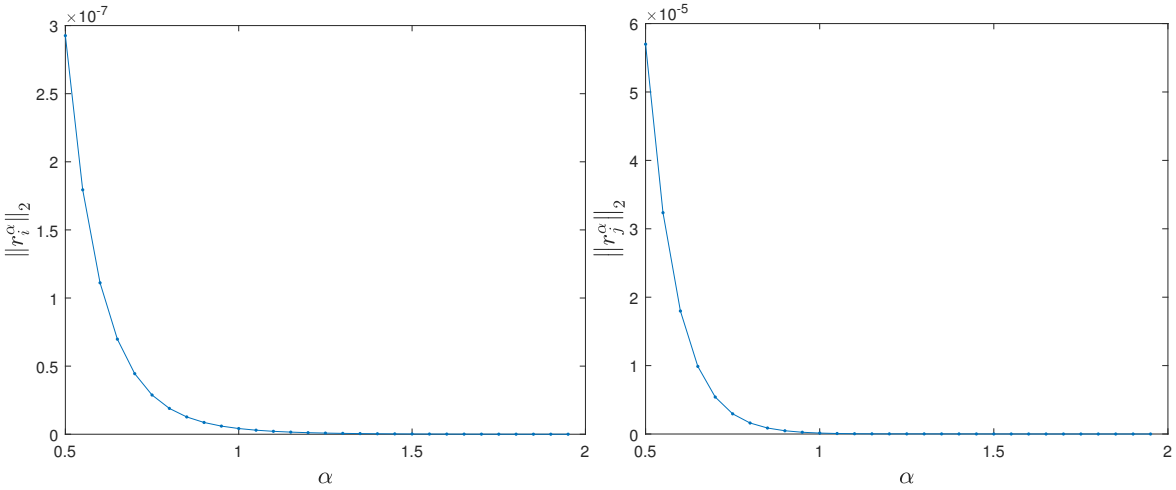


FIGURE 2.7: Left: norm of the residuals $\|r_i^\alpha\|_2$ that comes from the fitting $u_i \sim \zeta_1 {x_i}^{-\beta_1}$. Right: norm of the residuals $\|r_j^\alpha\|_2$ that comes from the fitting $u_j \sim \zeta_0 {x_j}^{-\beta_0}$.

If we consider a logarithmic scale in each asymptotic behavior:

$$\ln(u_i) \sim \ln(\zeta_1 / {x_i}^{\beta_1})$$

and

$$\ln(u_j) \sim \ln(\zeta_0 / {x_j}^{\beta_0}),$$

we obtain, equivalently,

$$\ln(u_i) \sim \ln(\zeta_1) - \beta_1 \ln(x_i)$$

and

$$\ln(u_j) \sim \ln(\zeta_0) - \beta_0 \ln(x_j).$$

Then, the coefficients $\beta_1$ and $\beta_0$ can be obtained and plotted against $\alpha$ as shown in Figure 2.8. On the left-hand side of Figure 2.8, we have plotted $\beta_1$ versus $\alpha$ when $x_i \ll 1$. On the right-hand side of Figure 2.8, we have plotted $\beta_0$ against $\alpha$ when $x_j \gg 1$. Both graphics show that $\beta$ is proportional to $\alpha$. In Figure 2.8, we have applied a linear fitting to every set of data and determine the slopes $\beta_1$ and $\beta_0$, which are 1.0115 and 1.0215, respectively.

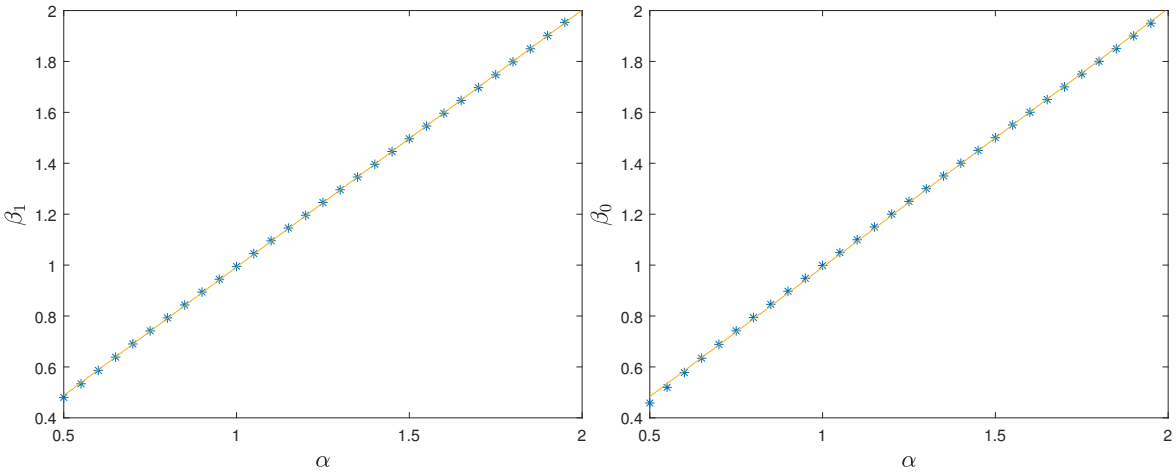


FIGURE 2.8: Left: $\beta_1$ values (against $\alpha$) corresponding to the fitting $\ln(u_i) \sim \ln(\zeta_1) - \beta_1 \ln(x_i)$, for those nodes $x_i \ll 1$. Right: $\beta_0$ values (against $\alpha$) corresponding to the fitting $\ln(u_j) \sim \ln(\zeta_0) - \beta_0 \ln(x_j)$, for those nodes $x_j \gg 1$.

For this reaction-diffusion model, we have done simulations with different time steps $\Delta t$, until $t = 20$. Hence, for a given $\Delta t$, $u_{num}(x, t, \Delta t)$ denotes the numerical approximation of $u(x, t)$. Since there is no explicit exact solution of $u(x, t)$, we will take as the "exact" solution $u_{num}(x, t, \Delta t_s)$, for a very small value of $\Delta t_s$, say, $\Delta t_s = 0.0001$. Thus, we are able to compute the convergence rate given by the formula

$$\mu(\alpha) = \lim_{m \to \infty} \log_2 \left( \frac{E(\Delta t)(\alpha)}{E(\Delta t/2)(\alpha)} \right),$$

where, for a given $\Delta t$, $E(\Delta t)$ is the global maximum error obtained through the discrete $L^\infty$-norm, i.e., we approximate the error as

$$E(\Delta t) \equiv \| u_{num}(x, t, \Delta t) - u_{num}(x, t, \Delta t_s) \|_\infty .$$

We have taken $\alpha = 1.80$ and $N = 256$; table 2.3 shows the evolution of $E(\Delta t)$ for different values of $\Delta t$. On the one hand, the errors are very small; on the other hand, the columns with $\log_2(E(\Delta t)/E(\Delta t/2))$ confirms clearly the fourth order of the Runge-Kutta scheme used to solve (1.11).

## 2.4 Future lines or research

A natural, important question is whether the method can be generalized to approximate numerically the fractional Laplacian in multiple dimensions. In what follows,

| $\Delta t$ | $E(\Delta t)$ | $\log_2\left(\frac{E(\Delta t)}{E(\Delta t/2)}\right)$ |
|---|---|---|
| 1/2 | $1.3701 \cdot 10^{-3}$ | 3.7068 |
| 1/4 | $1.0493 \cdot 10^{-4}$ | 3.8531 |
| 1/8 | $7.2609 \cdot 10^{-6}$ | 3.9264 |
| 1/16 | $4.7755 \cdot 10^{-7}$ | 3.9631 |
| 1/32 | $3.0619 \cdot 10^{-8}$ | 3.9815 |
| 1/64 | $1.9382 \cdot 10^{-9}$ | 3.9911 |
| 1/128 | $1.2189 \cdot 10^{-10}$ | |

TABLE 2.3: Error in discrete $L^\infty$-norm of $u_{num}$ at $t = 20$, for different values of $\Delta t$. Since the exact solution is not available, we use instead $u_{num}(x, 20, \Delta t_s)$ as reference.

we discuss a few lines of work.

We consider an equivalent expression of (1.1), which is the equivalent of (1.10) in $n$ dimensions (see [1]):

$$(-\Delta)^{\alpha/2} u(\vec{x}) = c_{n,\alpha} \int_{\mathbb{R}^n} \frac{u(\vec{x}) - u(\vec{x}+\vec{y})}{\|\vec{y}\|^{n+\alpha}} d\vec{y}, \tag{2.53}$$

where $\vec{x} \equiv (x_1, \ldots, x_n)$, $\vec{y} \equiv (y_1, \ldots, y_n)$, $\|\cdot\|$ denotes the Euclidean norm, and $c_{n,\alpha}$ is given by (1.2).

In order to extend the ideas in this thesis to $\mathbb{R}^n$, we need to express (2.53) in such a way that there is no subtraction in the numerator, as in Lemma 2.1.1. Since this is quite straightforward, we will show it below in the next subsection. The next step would be mapping the unbounded domain into a finite one, and the rank of possibilities is now richer than in the one-dimensional case. For instance, we could consider again $x_1 = L_1 \cot(s_1), \ldots, x_n = L_n \cot(s_n)$, which would map $\vec{x} \in \mathbb{R}^n$ into $\vec{s} = (s_1, \ldots, s_n) \in [0, \pi]^n$. Then, after extending $u(\vec{s})$, to $[0, 2\pi]^n$, we would approximate $u(\vec{s})$ using a pseudospectral approach, as in (2.14):

$$u(s) \approx \sum_{\vec{k} \in [-N, \ldots, N-1]^n} \hat{u}(\vec{k}) e^{i(\vec{k}\cdot\vec{s})}, \quad \vec{s} \in [0,\ 2\pi]^n;$$

so the problem would be reduced to computing the fractional Laplacian of $e^{i(\vec{k}\cdot\vec{s})}$, for which the lemmas bellow would be helpful. Another option would be to consider $n$-dimensional spherical coordinates [126]. For instance, in the two-dimensional case, this would mean working with polar coordinates, i.e., $\rho \in [0, \infty)$, $\theta \in [0, 2\pi]$. While working with $\theta$ would pose no problem, we could apply an algebraic map to to transform $[0, \infty)$ into a finite domain, etc. This approach might be especially useful if $u$ has radial symmetry. In any case, a detailed study of the multiple-dimensional case lies beyond the scope of this thesis, and we postpone it for the future.

### 2.4.1 Generalization of Lemma 2.1.1 to higher dimensions

We consider first the generalization to two dimensions.

**Lemma 2.4.1.** *Consider the twice continuous bounded function $u \in \mathcal{C}_b^2(\mathbb{R}^2)$, and such that $\lim_{\|(x,y)\|\to\infty} \|\nabla u(x,y)\| = 0$; then*

$$(-\Delta)^{\alpha/2}u(x,y) = -\frac{c_{2,\alpha}}{\alpha^2}\iint_{\mathbb{R}^2}\frac{\Delta u(p,q)}{\|(x-p,y-q)\|^{\alpha}}dp\,dq. \tag{2.54}$$

*Proof.* Let us take $n = 2$ in (2.53):

$$(-\Delta)^{\alpha/2}u(x,y) = c_{2,\alpha}\iint_{\mathbb{R}^2}\frac{u(x,y)-u(x+p,y+q)}{\|(p,q)\|^{2+\alpha}}dp\,dq. \tag{2.55}$$

We make a change of variable to polar coordinates in (2.55), i.e., $p = r\cos(\theta)$ and $q = r\sin(\theta)$; to simplify the notation, we write $c_\theta \equiv \cos(\theta)$, $s_\theta \equiv \sin(\theta)$. Then,

$$(-\Delta)^{\alpha/2}u(x,y) = -c_{2,\alpha}\int_0^{2\pi}\int_0^{\infty}\frac{u(x+c_\theta r, y+s_\theta r)-u(x,y)}{r^{1+\alpha}}dr\,d\theta.$$

Moreover, bearing in mind that

$$\frac{\partial}{\partial z}u(x+c_\theta z, y+s_\theta z) = c_\theta u_x(x+c_\theta z, y+s_\theta z) + s_\theta u_y(x+c_\theta z, y+s_\theta z),$$

we get

$$\begin{aligned}&(-\Delta)^{\alpha/2}u(x,y)\\ &\quad= -c_{2,\alpha}\int_0^{2\pi}\int_0^{\infty}\frac{1}{r^{1+\alpha}}\int_0^{r}[c_\theta u_x(x+c_\theta z, y+s_\theta z) + s_\theta u_y(x+c_\theta z, y+s_\theta z)]dz\,dr\,d\theta.\end{aligned}$$

Changing the order of integration,

$$\begin{aligned}(-\Delta)^{\alpha/2}u(x,y) &= -c_{2,\alpha}\int_0^{2\pi}\int_0^{\infty}\Big[(c_\theta u_x(x+c_\theta z, y+s_\theta z)\\ &\qquad + s_\theta u_y(x+c_\theta z, y+s_\theta z))\int_z^{\infty}\frac{1}{r^{1+\alpha}}dr\Big]dz\,d\theta\\ &= -\frac{c_{2,\alpha}}{\alpha}\int_0^{2\pi}\int_0^{\infty}\frac{1}{z^{\alpha}}\left[c_\theta u_x(x+c_\theta z, y+s_\theta z)\right.\\ &\qquad \left. + s_\theta u_y(x+c_\theta z, y+s_\theta z)\right]dz\,d\theta.\end{aligned}$$

Hence,

$$\begin{aligned}(-\Delta)^{\alpha/2}u(x,y) &= -\frac{c_{2,\alpha}}{\alpha}\int_0^{2\pi}\int_0^{\infty}\frac{r}{r^{2+\alpha}}\left[c_\theta r u_x(x+c_\theta r, y+s_\theta r)\right.\\ &\quad\left.+\, s_\theta r u_y(x+c_\theta r, y+s_\theta r)\right]dr\,d\theta\\ &= -\frac{c_{2,\alpha}}{\alpha}\iint_{\mathbb{R}^2}\frac{p\,u_x(x+p,y+q)+q\,u_y(x+p,y+q)}{\|(p,q)\|^{2+\alpha}}dp\,dq. \quad (2.56)\end{aligned}$$

Observe that this is precisely

$$(-\Delta)^{\alpha/2}u(x,y) = -\frac{c_{2,\alpha}}{\alpha}\iint_{\mathbb{R}^2}\frac{(p,q)\cdot\nabla u((x,y)+(p,q))}{\|(p,q)\|^{2+\alpha}}dp\,dq, \quad (2.57)$$

which is the two-dimensional equivalent of (2.7).

On the other hand,

$$\begin{aligned}\int_{\mathbb{R}}\frac{p\,u_x(x+p,y+q)}{(p^2+q^2)^{1+\alpha/2}}dp &= \frac{1}{\alpha}\int_{\mathbb{R}}\frac{u_{xx}(x+p,y+q)}{(p^2+q^2)^{\alpha/2}}dp,\\ \int_{\mathbb{R}}\frac{q\,u_y(x+p,y+q)}{(p^2+q^2)^{1+\alpha/2}}dq &= \frac{1}{\alpha}\int_{\mathbb{R}}\frac{u_{yy}(x+p,y+q)}{(p^2+q^2)^{\alpha/2}}dq,\end{aligned} \quad (2.58)$$

where we have integrated by parts, and used respectively

$$\lim_{p\to\pm\infty} u_x(x+p,y+q)\|(p,q)\|^{-\alpha} = 0,$$

and

$$\lim_{q\to\pm\infty} u_y(x+p,y+q)\|(p,q)\|^{-\alpha} = 0,$$

for all $q$ and $p$, respectively. Therefore, applying Fubini's theorem to (2.56), using (2.58), and making the change of variable $\tilde{p} = p - x$, $\tilde{q} = q - y$, we get (2.54), which concludes the proof.

□

*Remark.* The requirement that $\lim_{\|(x,y)\|\to\infty}\|\nabla u(x,y)\| = 0$ is sufficient for all $\alpha \in (0,2)$, but not sharp. A determination of the minimum requirements of $u$ is future work.

Lemma 2.4.1 can be immediately generalized to $n$ dimensions.

**Lemma 2.4.2.** *Consider the twice continuous bounded function $u \in C_b^n(\mathbb{R}^n)$, and such that $\lim_{\|\vec{x}\|\to\infty}\|\nabla u(\vec{x})\| = o(\|\vec{x}\|^{n-2})$; then*

$$(-\Delta)^{\alpha/2}u(\vec{x}) = -\frac{c_{n,\alpha}}{\alpha(n-2+\alpha)}\int_{\mathbb{R}^n}\frac{\Delta u(\vec{x}+\vec{y})}{\|\vec{y}\|^{n-2+\alpha}}d\vec{y}. \quad (2.59)$$

*Proof.* The proof is identical to that of Lemma 2.4.1. In this case, we use $n$-dimensional spherical coordinates [126]:

$$\begin{aligned} y_1 &= r\cos(\phi_1), \\ y_j &= r\cos(\phi_j)\prod_{k=1}^{j-1}\sin(\phi_k), \quad j \in \{2,\dots,n-2\}, \\ y_{n-1} &= r\sin(\theta)\cos(\phi_j)\prod_{k=1}^{n-2}\sin(\phi_k), \\ y_n &= r\cos(\theta)\cos(\phi_j)\prod_{k=1}^{n-2}\sin(\phi_k), \end{aligned}$$

with $r \in [0,\infty)$, $\theta \in [0,2\pi]$, and $\phi_j \in [0,\pi]$, for all $j$. Moreover, the Jacobian of the transformation is

$$J = r^{n-1}\prod_{k=1}^{n-2}\sin^k(\phi_{n-1-k}).$$

Therefore, bearing in mind that

$$\frac{d\vec{y}}{\|\vec{y}\|^{n+\alpha}} = \frac{1}{r^{1+\alpha}}\left(\prod_{k=1}^{n-2}\sin^k(\phi_{n-1-k})\right) dr\, d\phi_1 \dots d\phi_{n-2}\, d\theta,$$

and following exactly the same steps as in Lemma 2.4.1, we get

$$(-\Delta)^{\alpha/2}u(\vec{x}) = -\frac{c_{n,\alpha}}{\alpha}\int_{\mathbb{R}^n}\frac{\vec{y}\cdot\nabla u(\vec{x}+\vec{y})}{\|\vec{y}\|^{n+\alpha}}d\vec{y}, \tag{2.60}$$

which is the $n$-dimensional equivalent of (2.7) and (2.57).

Finally, we apply Fubini's theorem to (2.60) and integrate by parts each of the addends of the numerator. For instance,

$$\begin{aligned} &\int_{\mathbb{R}}\frac{y_1 u_{x_1}(\vec{x}+\vec{y})}{\|\vec{y}\|^{n+\alpha}}dy_1 \\ &\quad= \frac{1}{2}\int_{\mathbb{R}} u_{x_1}(x_1+y_1,\dots,x_n+y_n)2y_1(y_1^2+\dots+y_n^2)^{-n/2-\alpha/2}dy_1 \\ &\quad= -\frac{1}{-n+2-\alpha}\int_{\mathbb{R}} u_{x_1x_1}(x_1+y_1,\dots,x_n+y_n)(y_1^2+\dots+y_n^2)^{-n/2+1-\alpha/2}dy_1 \\ &\quad= \frac{1}{n-2+\alpha}\int_{\mathbb{R}}\frac{u_{x_1x_1}(\vec{x}+\vec{y})}{\|\vec{y}\|^{n-2+\alpha}}dy_1, \end{aligned}$$

where we have used

$$\lim_{y_1\to\pm\infty} u_{x_1}(\vec{x}+\vec{y})\|\vec{y}\|^{-n+2-\alpha} = 0,$$

for all $y_2,\dots,y_n$.

Putting everything together and writing the resulting expression as a convolution, we get (2.59), which concludes the proof.

□

*Remark.* The requirement

$$\lim_{\|\vec{x}\|\to\infty} \|\nabla u(\vec{x})\| = o(\|\vec{x}\|^{n-2})$$

is sufficient for all $\alpha \in (0,2)$, but not sharp. As already mentioned in the remark after the proof of Lemma 2.4.1, the determination of the minimum requirements of $u$ is future work.

# Chapter 3

# Numerical Approximation of the Fractional Laplacian on $\mathbb{R}$ Using Orthogonal Families

In this chapter, we obtain explicit expressions for the one-dimensional fractional Laplacian of the Higgins and Christov functions. We also explain how to implement these results efficiently in MATLAB [8], and give numerical examples as an application.

As in Chapter 2, we work with the definition (1.10) of the fractional Laplacian. Let us recall however (1.5), which is another definition of the fractional Laplacian consistent with (1.10) that associates the operator with the Fourier symbol:

$$\widehat{(-\Delta)^{\alpha/2}u}(\xi) = |\xi|^{\alpha}\hat{u}(\xi).$$

We observe that, when $\alpha = 0$, according to the Fourier symbol,

$$(-\Delta)^{0}u(x) = u(x).$$

We also observe, however, that, when the Fourier transform of $u(x)$ does not exist in the classical sense, the limit $\alpha \to 0^{+}$ is singular. For instance, take $u(x) = 1$, whose Fourier transform is the Dirac delta distribution, i.e., $\hat{u}(\xi) = 2\pi\delta(\xi)$, then

$$(-\Delta)^{\alpha}1 = \begin{cases} 1, & \alpha = 0, \\ 0, & \alpha > 0. \end{cases} \tag{3.1}$$

In Chapter 2, we have proposed a pseudospectral method to compute the fractional Laplacian of a bounded function $u(x)$ on $\mathbb{R}$ without truncation; and the main idea is to map $\mathbb{R}$ into the finite interval $[0, \pi]$ by means of the change of variable $x = L\cot(s)$, and then obtain the trigonometric Fourier series expansion of $u(x(s))$. Therefore, the central point of Chapter 2 is the efficient and accurate numerical computation of the fractional Laplacian of $e^{ins}$, for $n \in \mathbb{Z}$. Observe that the resulting expressions for $(-\Delta)^{1/2}e^{ins}$ are quite different, depending on whether $n$ is even or odd. At this point, a crucial observation is that the equivalent on $\mathbb{R}$ of the functions

$e^{i2ns}$ (i.e., the even case) are precisely the complex Higgins functions defined in [7] (see also [127, 128]):

$$\lambda_n(x) = \left(\frac{ix-1}{ix+1}\right)^n, \quad n \in \mathbb{Z}, \tag{3.2}$$

because $\lambda_n(\cot(s)) = e^{i2ns}$. Indeed, in this chapter, thanks to the structure of $\lambda_n(x)$, and using the definition (2.6):

$$(-\Delta)^{\alpha/2}u(x) = \frac{c_\alpha}{\alpha}\int_0^\infty \frac{u_x(x-y) - u_x(x+y)}{y^\alpha}dy, \tag{3.3}$$

we obtain an explicit expression of their fractional Laplacian by means of contour integration:

$$(-\Delta)^{\alpha/2}\lambda_n(x) = \begin{cases} 0, & n = 0, \\ -\dfrac{2|n|\Gamma(1+\alpha)}{(i\,\mathrm{sgn}(n)x+1)^{1+\alpha}} & \\ \quad {}_2F_1\left(1-|n|, 1+\alpha; 2; \dfrac{2}{i\,\mathrm{sgn}(n)x+1}\right), & n \in \mathbb{Z}\backslash\{0\}, \end{cases} \tag{3.4}$$

where ${}_2F_1$ is the Gaussian hypergeometric function (see for instance [4, Ch. 15]). From this result, we also derive several other related ones outlined below.

The structure of this chapter is as follows. In Section 3.1, we prove (3.4), which constitutes the main result. We observe that $\{\lambda_n(x)\}$ is a complete orthogonal system in $\mathcal{L}^2(\mathbb{R})$ with weight $w(x) = 1/(\pi(1+x^2))$, because $\{e^{i2ns}\}$ is a complete orthonormal system in $\mathcal{L}^2([0,\pi])$, normalized by $w = 1/\pi$. Therefore, the related family of functions

$$\mu_n(x) = \frac{(ix-1)^n}{(ix+1)^{n+1}}, \quad n \in \mathbb{Z}, \tag{3.5}$$

known as the complex Christov functions [7, 128–130], form a complete orthogonal system in $\mathcal{L}^2(\mathbb{R})$ (normalized by the factor $1/\pi$).

Throughout this chapter, we use the definition and notation from [7] for the following families of functions:

- Cosine-like Higgins functions:

$$\mathrm{CH}_{2n}(x) = \frac{\lambda_n(x) + \lambda_{-n}(x)}{2}, \quad n = 0, 1, 2, \ldots \tag{3.6}$$

- Sine-like Higgins functions:

$$\mathrm{SH}_{2n+1}(x) = \frac{\lambda_{n+1}(x) - \lambda_{-n-1}(x)}{2i}, \quad n = 0, 1, 2, \ldots \tag{3.7}$$

- Cosine-like Christov functions:

$$\mathrm{CC}_{2n}(x) = \frac{\mu_n(x) - \mu_{-n-1}(x)}{2}, \quad n = 0, 1, 2, \ldots \tag{3.8}$$

- Sine-like Christov functions:

$$\mathrm{SC}_{2n+1}(x) = -\frac{\mu_n(x) + \mu_{-n-1}(x)}{2i}, \quad n = 0, 1, 2, \ldots \tag{3.9}$$

In Section 3.2, starting from (3.4), we calculate the fractional Laplacian of (3.5)-(3.9).

Even if the fractional Laplacian of all the families considered here can be computed accurately with the technique explained in Chapter 2, expressions like (3.4) have the advantage of being very compact and, hence, it is effortless to use them in numerical applications, provided that fast accurate implementations of the Gaussian hypergeometric function ${}_2F_1$ are available. Therefore, in Section 3.3, using MATLAB, we test their adequacy from a numerical point of view, comparing the numerical results with those in Chapter 2. On the one hand, for moderately large values of $n$, the use of variable precision arithmetic seems unavoidable; on the other hand, our implementation of ${}_2F_1$ largely outperforms that of MATLAB. Finally, even if the method in Chapter 2 is faster, the method developed in this chapter is much easier to implement and still competitive for not too large values of $n$.

## 3.1 Fractional Laplacian of the complex Higgins functions

Before we proceed, let us recall some well-known definitions. Given $z \in \mathbb{C}$, the generalized binomial coefficient is defined by

$$\binom{z}{n} = \begin{cases} \dfrac{z(z-1)\ldots(z-n+1)}{n!}, & n \in \mathbb{N}, \\ 1, & n = 0. \end{cases}$$

where $\mathbb{N} = \{1, 2, 3, \ldots\}$. Therefore, if $z$ is a nonnegative integer, and $n > z$, then $\binom{z}{n} = 0$. Furthermore, it is immediate to check that, for all $z$ and $n$,

$$\binom{z}{n} = (-1)^n \binom{n-1-z}{n}.$$

We will also need the Pochhammer symbol, which represents the rising factorial, and is defined by

$$(z)_n = \begin{cases} z(z+1)\ldots(z+n-1), & n \in \mathbb{N}, \\ 1, & n = 0. \end{cases}$$

Observe that, when $z$ is not zero or a negative integer, an equivalent definition is

$$(z)_n = \frac{\Gamma(z+n)}{\Gamma(z)};$$

in particular, when $z \in \mathbb{N}$,

$$(z)_n = \frac{(z+n-1)!}{(z-1)!}.$$

Remark that, if $z$ is a negative integer or zero, and $n > |z|$, then $(z)_n = 0$. Moreover, the following identities will be useful, too:

$$\begin{aligned}(-z)_n &= (-1)^n (z-n+1)_n, \\ \binom{z}{n} &= \frac{(z-n+1)_n}{n!} = \frac{(-1)^n (-z)_n}{n!}.\end{aligned}$$

The Pochhammer symbol also appears in the definition of the Gaussian hypergeometric function ${}_2F_1$ (see for instance [4, Ch. 15]). Let $a, b, c, z \in \mathbb{C}$; then, ${}_2F_1$ is defined by

$${}_2F_1(a,b;c;z) = \sum_{k=0}^{\infty} \frac{(a)_k (b)_k}{(c)_k} \frac{z^k}{k!}. \tag{3.10}$$

In general, the infinite series converges for $|z| < 1$. However, in our case, we take $a$ to be a negative integer, so the sum is finite, because of the properties of the Pochhammer symbol. More precisely, in this chapter, we are interested in the following two particular cases:

$${}_2F_1(-m, 1+\alpha; 1; z) = \sum_{k=0}^{m} \binom{m}{k} \binom{-1-\alpha}{k} z^k, \tag{3.11}$$

$$\begin{aligned}{}_2F_1(-m, 1+\alpha; 2; z) &= -\frac{1}{\alpha} \sum_{k=0}^{m} \binom{m}{k} \binom{-\alpha}{k+1} z^k \\ &= \frac{1}{m+1} \sum_{k=0}^{m} \binom{m+1}{k+1} \binom{-1-\alpha}{k} z^k,\end{aligned} \tag{3.12}$$

for $m \in \mathbb{N}$. Observe that the identities also hold after replacing $k$ by $m-k$ in the sums, a fact that we will also use below. On the other hand, if $m = 0$,

$${}_2F_1(0, 1+\alpha; 1; z) = {}_2F_1(0, 1+\alpha; 2; z) = 1.$$

Bearing in mind the previous arguments, let us prove (3.4), which is the main result of this chapter. Note that we work with the binomial coefficient rather than with the Pochhammer symbol, because we think that the former is more intuitive.

**Theorem 3.1.1.** *Let $\lambda_n(x)$ be defined as in* (3.2)*. Then, $(-\Delta)^{\alpha/2} \lambda_n(x)$ is given by* (3.4)*.*

*Proof.* The case with $n = 0$ is trivial, because $\lambda_0(x) = 1$. Assume $n \in \mathbb{N}$. The derivative of (3.2) is

$$\lambda_n'(x) = -\frac{2ni}{(x-i)^2}\left(\frac{x+i}{x-i}\right)^{n-1}.$$

Introducing this expression in (3.3), we get

$$(-\Delta)^{\alpha/2}\lambda_n(x) = 2ni\frac{2^{\alpha-1}\Gamma(1/2+\alpha/2)}{\sqrt{\pi}\Gamma(1-\alpha/2)}\int_0^\infty g_n(y;x)dy, \tag{3.13}$$

with integrand

$$g_n(z;x) = \frac{(z+x+i)^{n-1}(z-x+i)^{n+1} - (z-x-i)^{n-1}(z+x-i)^{n+1}}{z^\alpha(z+x-i)^{n+1}(z-x+i)^{n+1}},$$

where, in this notation, we regard $x$ as a parameter, rather than as an independent variable.

We next compute (3.13) for every $x$, by integrating $g_n(z;x)$ along certain integration contours $C$ in $\mathbb{C}$, and using Cauchy's integral theorem. Since

$$z^\alpha = e^{\alpha\ln(z)} = e^{\alpha(\ln(z)+i\arg(z))},$$

$z^\alpha$ has a branch cut. In what follows, we consider the principal branch of the logarithm, which corresponds to $-\pi < \arg(z) \leq \pi$; in particular, $(-1)^\alpha = e^{i\pi\alpha}$, unless the branch cut is crossed. The branch choice determines also how we choose the contours.

In Figure 3.1, we have depicted one such contour $C$ for $x > 0$, which consists of four parts, avoids the branch cut, but encloses the poles $z = i - x$ and $z = x - i$, for every $x$. Then, by the residue theorem, the integral along it is equal to the sum of the residues. The pieces of the contour that run parallel to the branch cut will give the approximation of the integral from 0 to $\infty$; the other pieces will give integrals that tend to zero, when $C$ tends to one contour that encloses $\mathbb{C}$, except for the brach cut. More precisely, for every $x \in \mathbb{R}$, we take $R > 0$, such that $R \gg (1+|x|^2)^{1/2}$, and $r > 0$, such that $r \ll (1+|x|^2)^{1/2}$, for instance, $r = 1/R$. We also take $\delta > 0$, such that, with $\theta_1 \in (-\pi, -\pi/2)$ and $\theta_2 \in (\pi/2, \pi)$ fixed, $\delta = r\sin\theta_2$. Then, we define

$$C = C_1 \cup C_R \cup C_2 \cup C_r,$$

with

$$\begin{aligned}
C_1 &= \{-y - i\delta : y \in (-r\cos\theta_1, -R\cos\theta_1)\},\\
C_R &= \{Re^{\theta i} : \theta \in (\theta_1, \theta_2)\},\\
C_2 &= \{y + i\delta : y \in (R\cos\theta_2, r\cos\theta_2)\},\\
C_r &= \{re^{-\theta i} : \theta \in (-\theta_2, -\theta_1)\}.
\end{aligned}$$

Here, we have omitted the dependecy on $x$ for simplicity of notation. Now, by Cauchy's residue theorem, we have:

$$\begin{aligned}\int_C g_n(z;x)dz &= \int_{C_1} g_n(z;x)dz + \int_{C_r} g_n(z;x)dz \\ &\quad + \int_{C_2} g_n(z;x)dz + \int_{C_R} g_n(z;x)dz \\ &= 2\pi i[\mathrm{Res}(g_n(z;x), i-x) + \mathrm{Res}(g_n(z;x), x-i)]. \end{aligned} \tag{3.14}$$

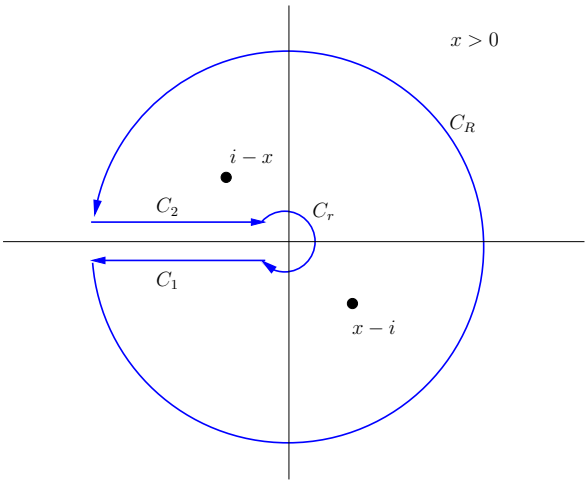


FIGURE 3.1: An example of integration contour, for $x > 0$.

In order to compute the residues of $g_n(z;x)$ at $z = i - x$ and $z = x - i$, we use the general Leibniz rule:

$$\frac{d^n}{dz^n}(p(z)q(z)) = \sum_{k=0}^{n} \binom{n}{k} p^{(k)}(z) q^{(n-k)}(z).$$

In this way, we obtain

$$\begin{aligned}&\mathrm{Res}(g_n(z;x), i-x) \\ &= \frac{1}{n!}\frac{d^n}{dz^n}\left(\frac{(z+x+i)^{n-1}(z-x+i)^{n+1} - (z-x-i)^{n-1}(z+x-i)^{n+1}}{z^\alpha (z-x+i)^{n+1}}\right)\Bigg|_{z=i-x} \\ &= \frac{1}{n!}\frac{d^n}{dz^n} z^{-\alpha}(z+x+i)^{n-1}\Big|_{z=i-x} \\ &= \frac{1}{n!}\sum_{k=1}^{n}\binom{n}{k}\left[k!\binom{-\alpha}{k} z^{-\alpha-k}\right]\left[\frac{(n-1)!}{(n-1-(n-k))!}(z+x+i)^{n-1-(n-k)}\right]\Bigg|_{z=i-x} \\ &= \sum_{k=1}^{n}\binom{n-1}{k-1}\binom{-\alpha}{k}(i-x)^{-\alpha-k}(2i)^{k-1}. \end{aligned} \tag{3.15}$$

Likewise, we have

$$\operatorname{Res}(g_n(z;x), x-i) = -\sum_{k=1}^{n} \binom{n-1}{k-1} \binom{-\alpha}{k} (x-i)^{-\alpha-k} (-2i)^{k-1}. \tag{3.16}$$

In order to compute (3.14), we observe that the first and third integrals tend to zero, as $R$ tends to infinity (this can be easily seen by changing to polar coordinates and recalling that $r = 1/R$). In what regards the first and third integrals, we have that $C_1$ tends to $(-\infty, 0)$, parameterized by $-y$, with $y \in (0, \infty)$, and $C_2$ tends to $(-\infty, 0)$, parameterized by $y$, with $y \in (-\infty, 0)$. We observe that $g_n(-y;x) = (-1)^{1-\alpha} g_n(y;x)$, where the argument of $-1$ is determined by the curve $C_i$ before taking the limit $R \to \infty$. Thus, it is $-\pi$ on $C_1$, and $\pi$ on $C_2$. Then,

$$\lim_{R\to\infty} \int_{C_1} g_n(z;x)dz = -\int_0^\infty g_n(-y;x)dy = e^{i\pi\alpha} \int_0^\infty g_n(y;x)dy,$$

and

$$\begin{aligned}\lim_{R\to\infty} \int_{C_2} g_n(z;x)dz &= \int_{-\infty}^0 g_n(y;x)dy = \int_0^\infty g_n(-y;x)dy \\ &= -e^{-i\pi\alpha} \int_0^\infty g_n(y;x)dy.\end{aligned}$$

Therefore,

$$\begin{aligned}\int_C g_n(z;x)dz &= (e^{i\pi\alpha} - e^{-i\pi\alpha}) \int_0^\infty g_n(y;x)dy \\ &= 2\pi i[\operatorname{Res}(g_n(z;x), i-x) + \operatorname{Res}(g_n(z;x), x-i)].\end{aligned}$$

Finally, after some manipulations, we have

$$\begin{aligned}\int_0^\infty g_n(y;x)dy &= \frac{2\pi i}{2i\sin(\pi\alpha)}[\operatorname{Res}(g_n(z;x), i-x) + \operatorname{Res}(g_n(z;x), x-i)] \\ &= \frac{\pi}{\sin(\pi\alpha)} \sum_{k=1}^{n} \binom{n-1}{k-1} \binom{-\alpha}{k} [(i-x)^{-\alpha-k}(2i)^{k-1} \\ &\qquad - (x-i)^{-\alpha-k}(-2i)^{k-1}] \\ &= -\frac{\pi((-1)^{-\alpha}+1)}{\sin(\pi\alpha)} \sum_{k=1}^{n} \binom{n-1}{k-1} \binom{-\alpha}{k} (x-i)^{-\alpha-k}(-2i)^{k-1} \\ &= -\frac{\pi(e^{-i\pi\alpha}+1)}{\sin(\pi\alpha)(x-i)^{1+\alpha}} \sum_{k=0}^{n-1} \binom{n-1}{k} \binom{-\alpha}{k+1} \left(\frac{-2i}{x-i}\right)^k \\ &= \frac{\pi\alpha e^{-i\pi\alpha/2}(i)^{1+\alpha}}{\sin(\pi\alpha/2)(ix+1)^{1+\alpha}} {}_2F_1\left(1-n, 1+\alpha, 2, \frac{2}{ix+1}\right),\end{aligned}$$

where we have used (3.12). Substituting this last expression into (3.13), and using Euler's reflection formula,

$$\Gamma(w)\Gamma(1-w) = \frac{\pi}{\sin(\pi w)},$$

at $w = \alpha/2$, we obtain

$$(-\Delta)^{\alpha/2}\lambda_n(x) = 2ni\frac{2^{\alpha-1}\Gamma(1/2+\alpha/2)}{\sqrt{\pi}\Gamma(1-\alpha/2)}\frac{i\alpha\Gamma(\alpha/2)\Gamma(1-\alpha/2)}{(ix+1)^{1+\alpha}} \\ {}_2F_1\left(1-n, 1+\alpha, 2, \frac{2}{ix+1}\right). \tag{3.17}$$

Then, (3.4) for $n \in \mathbb{N}$ follows from (3.17), using Legendre's duplication formula

$$\Gamma(w)\Gamma\left(w+\frac{1}{2}\right) = 2^{1-2w}\sqrt{\pi}\Gamma(2w),$$

at $w = \alpha/2$. In order to finish the proof, we notice that the case where $n$ is a negative integer follows from the symmetry

$$\lambda_n(x) = \overline{\lambda_{-n}(x)}, \quad n \in \mathbb{N}. \tag{3.18}$$

□

## 3.2 Fractional Laplacian of other families of functions

In this section, we compute the fractional Laplacian of the families of functions defined by (3.5)-(3.9). Let us obtain first the fractional Laplacian of the complex Christov Functions:

**Proposition 3.2.1.** *Let $\mu_n(x)$ be defined as in* (3.5)*. Then,*

$$(-\Delta)^{\alpha/2}\mu_n(x) \\ = \begin{cases} \dfrac{\Gamma(1+\alpha)}{(ix+1)^{1+\alpha}}{}_2F_1\left(-n, 1+\alpha; 1; \dfrac{2}{ix+1}\right), & n = 0, 1, 2, \dots, \\ -\dfrac{\Gamma(1+\alpha)}{(-ix+1)^{1+\alpha}}{}_2F_1\left(1+n, 1+\alpha; 1; \dfrac{2}{-ix+1}\right), & n = -1, -2, -3, \dots \end{cases} \tag{3.19}$$

*Moreover, the expression for $n = 0$ reduces to*

$$(-\Delta)^{\alpha/2}\mu_0(x) = \frac{\Gamma(1+\alpha)}{(ix+1)^{1+\alpha}}.$$

*Proof.* First, we note that $\mu_n(x)$ can be expressed in terms of $\lambda_n(x)$ as follows:

$$\mu_n(x) = \lambda_n(x)\frac{1}{(ix+1)} = \frac{1}{2}\lambda_n(x)\frac{(ix+1)-(ix-1)}{(ix+1)} = \frac{\lambda_n(x)-\lambda_{n+1}(x)}{2};$$

therefore,

$$(-\Delta)^{\alpha/2}\mu_n(x) = \frac{(-\Delta)^{\alpha/2}\lambda_n(x) - (-\Delta)^{\alpha/2}\lambda_{n+1}(x)}{2}, \tag{3.20}$$

so we just have to use (3.4) and simplify the resulting expression. First we assume that $n \in \mathbb{N}$. Substituting (3.4) into this last equation, and using

$$\binom{n+1}{k+1} = \binom{n}{k+1} + \binom{n}{k},$$

we get

$$\begin{aligned}(-\Delta)^{\alpha/2}\mu_n(x) = \frac{\Gamma(1+\alpha)}{(ix+1)^{1+\alpha}} &\left[\sum_{k=0}^{n}\binom{n+1}{k+1}\binom{-1-\alpha}{k}\left(\frac{2}{ix+1}\right)^k\right.\\ &\left.-\sum_{k=0}^{n-1}\binom{n}{k+1}\binom{-1-\alpha}{k}\left(\frac{2}{ix+1}\right)^k\right]\\ = \frac{\Gamma(1+\alpha)}{(ix+1)^{1+\alpha}}&\sum_{k=0}^{n}\binom{n}{k}\binom{-1-\alpha}{k}\left(\frac{2}{ix+1}\right)^k,\end{aligned}$$

which is (3.19), for $n \in \mathbb{N}$.

On the other hand, when $n = 0$, again from (3.4),

$$\begin{aligned}(-\Delta)^{\alpha/2}\mu_0(x) &= -\frac{(-\Delta)^{\alpha/2}\lambda_1(x)}{2}\\ &= \frac{\Gamma(1+\alpha)}{(ix+1)^{1+\alpha}}{}_2F_1\left(0, 1+\alpha; 2; \frac{2}{ix+1}\right)\\ &= \frac{\Gamma(1+\alpha)}{(ix+1)^{1+\alpha}},\end{aligned}$$

and (3.19) also holds.

Finally, when $n$ is a negative integer, we observe that

$$\mu_n(x) = -\overline{\mu_{-1-n}(x)}, \tag{3.21}$$

so, in that case, we use

$$(-\Delta)^{\alpha/2}\mu_n(x) = -\overline{(-\Delta)^{\alpha/2}\mu_{-1-n}(x)},$$

which concludes the proof.

□

Before we compute the fractional Laplacian of the cosine-like and sine-like Higgins functions (3.6) and (3.7), we first express (3.4) as a polynomial on $x$ times a negative power of $(i\,\mathrm{sgn}(n)x+1)$.

**Lemma 3.2.1.** *Let $\lambda_n(x)$ be defined as in* (3.2)*, and $n \in \mathbb{Z}\backslash\{0\}$. Then,* (3.4) *can be expanded as*

$$(-\Delta)^{\alpha/2}\lambda_n(x) = -\frac{2|n|\Gamma(1+\alpha)}{(i\,\mathrm{sgn}(n)x+1)^{|n|+\alpha}} \sum_{l=0}^{|n|-1} \binom{|n|-1}{l} (i\,\mathrm{sgn}(n)x)^{|n|-1-l} \\ {}_2F_1(-l, 1+\alpha; 2; 2). \tag{3.22}$$

*Proof.* Assume first that $n \in \mathbb{N}$. We express ${}_2F_1$ in (3.17) as a sum; then, replacing the index $k$ by $n-1-k$ in the sum, and expanding $(ix+1)^l$ by Newton's binomial formula, we get

$$\begin{aligned}(-\Delta)^{\alpha/2}\lambda_n(x) &= -\frac{2\Gamma(1+\alpha)}{(ix+1)^{1+\alpha}} \sum_{k=0}^{n-1} \binom{n}{k+1} \binom{-1-\alpha}{k} \left(\frac{2}{ix+1}\right)^k \\ &= -\frac{2^n\Gamma(1+\alpha)}{(ix+1)^{n+\alpha}} \sum_{k=0}^{n-1} \sum_{l=0}^{k} \frac{1}{2^k} \binom{n}{k} \binom{k}{l} \binom{-1-\alpha}{n-1-k} (ix)^l. \end{aligned} \tag{3.23}$$

We now interchange the order of the sums and replace the indices $l$ by $n-1-l$ and $k$ by $n-1-k$; then, after some rewriting, we get,

$$\begin{aligned}&(-\Delta)^{\alpha/2}\lambda_n(x) \\ &= -\frac{2^n\Gamma(1+\alpha)}{(ix+1)^{n+\alpha}} \sum_{l=0}^{n-1} (ix)^l \sum_{k=l}^{n-1} \frac{1}{2^k} \binom{n}{k} \binom{k}{l} \binom{-1-\alpha}{n-1-k} \\ &= -\frac{2^n\Gamma(1+\alpha)}{(ix+1)^{n+\alpha}} \sum_{l=0}^{n-1} (ix)^{n-1-l} \sum_{k=0}^{l} \frac{1}{2^{n-1-k}} \binom{n}{n-1-k} \binom{n-1-k}{n-1-l} \binom{-1-\alpha}{k} \\ &= -\frac{2n\Gamma(1+\alpha)}{(ix+1)^{n+\alpha}} \sum_{l=0}^{n-1} \binom{n-1}{l} (ix)^{n-1-l} \frac{1}{l+1} \sum_{k=0}^{l} \binom{l+1}{k+1} \binom{-1-\alpha}{k} 2^k, \end{aligned} \tag{3.24}$$

which yields (3.22). The case with $n$ a negative integer follows from (3.18).

□

*Remark.* If we only consider $l = 0$ in (3.22), we get the asymptotic behavior of $(-\Delta)^{\alpha/2}\lambda_n(x)$:

$$(-\Delta)^{\alpha/2}\lambda_n(x) \sim -\frac{2|n|\Gamma(1+\alpha)(i\,\mathrm{sgn}(n)x)^{|n|-1}}{(i\,\mathrm{sgn}(n)x+1)^{|n|+\alpha}} \sim -\frac{2|n|\Gamma(1+\alpha)}{(i\,\mathrm{sgn}(n)x)^{1+\alpha}}, \quad x \to \pm\infty,$$

i.e., $|(-\Delta)^{\alpha/2}\lambda_n(x)|$ decays at infinity as $|x|^{-1-\alpha}$ for all $n$, whereas

$$\lambda_n(x) - 1 = \left(1 - \frac{2}{ix+1}\right)^n - 1 \sim -\frac{2n}{ix+1} \sim -\frac{2n}{ix}, \quad x \to \pm\infty,$$

i.e., $|\lambda_n(x) - 1|$ decays at infinity as $|x|^{-1}$, for all $n$. Therefore, the operator $(-\Delta)^{\alpha/2}$ introduces an extra decay of $|x|^{-\alpha}$.

As a consequence of Lemma 3.2.1, we also have the following result.

**Proposition 3.2.2.** *The fractional Laplacian of* (3.6) *and* (3.7) *is given respectively by*

$$(-\Delta)^{\alpha/2}\,\mathrm{CH}_{2n}(x) = \begin{cases} 0, & n = 0, \\ -\dfrac{2\Gamma(1+\alpha)}{(1+x^2)^{(1+\alpha)/2}} \sin\left((1+\alpha)\operatorname{arccot}(x) - \dfrac{\pi\alpha}{2}\right), & n = 1, \\ -\dfrac{2n\Gamma(1+\alpha)}{(1+x^2)^{(n+\alpha)/2}} \sin\left((n+\alpha)\operatorname{arccot}(x) - \dfrac{\pi\alpha}{2}\right) & \\ \quad \cdot \displaystyle\sum_{l=0}^{\lfloor \frac{n-1}{2} \rfloor} (-1)^l \binom{n-1}{2l} x^{n-1-2l}\, {}_2F_1(-2l, 1+\alpha; 2; 2) & \\ \quad + \dfrac{2n\Gamma(1+\alpha)}{(1+x^2)^{(n+\alpha)/2}} \cos\left((n+\alpha)\operatorname{arccot}(x) - \dfrac{\pi\alpha}{2}\right) & \\ \quad \cdot \displaystyle\sum_{l=0}^{\lfloor \frac{n-2}{2} \rfloor} (-1)^l \binom{n-1}{2l+1} x^{n-2-2l}\, {}_2F_1(-2l-1, 1+\alpha; 2; 2), & n = 2, 3, 4, \ldots, \end{cases} \tag{3.25}$$

*and*

$$(-\Delta)^{\alpha/2}\,\mathrm{SH}_{2n+1}(x) = \begin{cases} \dfrac{2\Gamma(1+\alpha)}{(1+x^2)^{(1+\alpha)/2}} \cos\left((1+\alpha)\operatorname{arccot}(x) - \dfrac{\pi\alpha}{2}\right), & n = 0, \\ \dfrac{2(n+1)\Gamma(1+\alpha)}{(1+x^2)^{(n+1+\alpha)/2}} \cos\left((n+1+\alpha)\operatorname{arccot}(x) - \dfrac{\pi\alpha}{2}\right) & \\ \quad \cdot \displaystyle\sum_{l=0}^{\lfloor \frac{n}{2} \rfloor} (-1)^l \binom{n}{2l} x^{n-2l}\, {}_2F_1(-2l, 1+\alpha; 2; 2) & \\ \quad + \dfrac{2n\Gamma(1+\alpha)}{(1+x^2)^{(n+1+\alpha)/2}} \sin\left((n+1+\alpha)\operatorname{arccot}(x) - \dfrac{\pi\alpha}{2}\right) & \\ \quad \cdot \displaystyle\sum_{l=0}^{\lfloor \frac{n-1}{2} \rfloor} (-1)^l \binom{n}{2l+1} x^{n-1-2l}\, {}_2F_1(-2l-1, 1+\alpha; 2; 2), & n = 1, 2, 3, \ldots. \end{cases} \tag{3.26}$$

*Proof.* From the definition of (3.6) and (3.7), together with (3.18) and Euler's formula

$$e^{is} = \cos(s) + i\sin(s),$$

we have immediately, for $n = 0, 1, 2, \ldots$,

$$\mathrm{CH}_{2n}(x) = \Re(\lambda_n(x)), \quad \mathrm{SH}_{2n}(x) = \Im(\lambda_{n+1}(x)),$$

and this implies (see also [7]) that

$$\begin{aligned}\mathrm{CH}_{2n}(\cot(s)) &= \cos(2ns),\\ \mathrm{SH}_{2n+1}(\cot(s)) &= \sin(2(n+1)s).\end{aligned}$$

Since $\mathrm{CH}_{2n}(x)$ and $\mathrm{SH}_{2n}(x)$ are real, their fractional Laplacian is also real. We thus want to obtain expressions of $(-\Delta)^{\alpha/2}\,\mathrm{CH}_{2n}(x)$ and $(-\Delta)^{\alpha/2}\,\mathrm{SH}_{2n}(x)$ that avoid the use of complex numbers. To do this, we express all the complex numbers in (3.24) in their polar form: e.g., $i^\alpha = e^{i\pi\alpha/2}$,

$$\frac{1}{(x-i)^{n+\alpha}} = \frac{(x+i)^{n+\alpha}}{(1+x^2)^{n+\alpha}} = \frac{e^{i(n+\alpha)\,\mathrm{arccot}(x)}}{(1+x^2)^{(n+\alpha)/2}},$$

etc. Note that, since $x = \cot(s)$, with $s \in [0, \pi]$, we have chosen the definitions of the arctangent and the arccotangent such that $\arctan(1/x) = \mathrm{arccot}(x) \in [0, \pi]$ in the last expression. Then, (3.24) becomes

$$\begin{aligned}(-\Delta)^{\alpha/2}\lambda_n(x) = -\frac{2n\Gamma(1+\alpha)}{(1+x^2)^{(n+\alpha)/2}} \sum_{l=0}^{n-1} \binom{n-1}{l} x^{n-1-l}\\ e^{i((n+\alpha)\,\mathrm{arccot}(x) - \pi(l+1+\alpha)/2)}\,{}_2F_1(-l, 1+\alpha; 2; 2). \qquad (3.27)\end{aligned}$$

With respect to (3.25), the case $n = 0$ is trivial, because $\mathrm{CH}_0(1) = 1$; otherwise, taking the real part of (3.27),

$$\begin{aligned}(-\Delta)^{\alpha/2}\,\mathrm{CH}_{2n}(x) &= -\frac{2n\Gamma(1+\alpha)}{(1+x^2)^{(n+\alpha)/2}} \sum_{l=0}^{n-1} \binom{n-1}{l} x^{n-1-l}\\ &\qquad \sin\left((n+\alpha)\,\mathrm{arccot}(x) - \frac{\pi\alpha}{2} - \frac{\pi l}{2}\right) {}_2F_1(-l, 1+\alpha; 2; 2)\\ &= -\frac{2n\Gamma(1+\alpha)}{(1+x^2)^{(n+\alpha)/2}} \sin\left((n+\alpha)\,\mathrm{arccot}(x) - \frac{\pi\alpha}{2}\right)\\ &\qquad \sum_{l=0}^{n-1} \cos\left(\frac{\pi l}{2}\right) \binom{n-1}{l} x^{n-1-l}\,{}_2F_1(-l, 1+\alpha; 2; 2)\\ &\quad + \frac{2n\Gamma(1+\alpha)}{(1+x^2)^{(n+\alpha)/2}} \cos\left((n+\alpha)\,\mathrm{arccot}(x) - \frac{\pi\alpha}{2}\right)\\ &\qquad \sum_{l=0}^{n-1} \sin\left(\frac{\pi l}{2}\right) \binom{n-1}{l} x^{n-1-l}\,{}_2F_1(-l, 1+\alpha; 2; 2).\end{aligned}$$

When $n = 1$, the first sum is equal to one, and the second one is equal to zero. For the other values of $n$, observe that $\cos(\pi l/2)$ is zero if and only if $l$ is odd, whereas $\sin(\pi l/2)$ is zero, if and only if $l$ is even. Therefore, substituting $l$ by $2l$ in the first term, and by $2l+1$ in the second term, the proof of (3.25) is concluded.

Likewise, taking the imaginary part of (3.27), and replacing $n$ by $n+1$:

$$\begin{aligned}(-\Delta)^{\alpha/2}\,\mathrm{SH}_{2n+1}(x) &= \frac{2(n+1)\Gamma(1+\alpha)}{(1+x^2)^{(n+1+\alpha)/2}}\sum_{l=0}^{n}\binom{n}{l}x^{n-l}\\ &\qquad \cos\left((n+1+\alpha)\operatorname{arccot}(x)-\frac{\pi\alpha}{2}-\frac{\pi l}{2}\right){}_2F_1(-l,1+\alpha;2;2)\\ &= \frac{2(n+1)\Gamma(1+\alpha)}{(1+x^2)^{(n+1+\alpha)/2}}\cos\left((n+1+\alpha)\operatorname{arccot}(x)-\frac{\pi\alpha}{2}\right)\\ &\quad \sum_{l=0}^{n}\cos\left(\frac{\pi l}{2}\right)\binom{n}{l}x^{n-l}{}_2F_1(-l,1+\alpha;2;2)\\ &\quad +\frac{2(n+1)\Gamma(1+\alpha)}{(1+x^2)^{(n+1+\alpha)/2}}\sin\left((n+1+\alpha)\operatorname{arccot}(x)-\frac{\pi\alpha}{2}\right)\\ &\quad \sum_{l=0}^{n}\sin\left(\frac{\pi l}{2}\right)\binom{n}{l}x^{n-l}{}_2F_1(-l,1+\alpha;2;2).\end{aligned}$$

When $n = 0$, the first sum is equal to one, and the second one is equal to zero. For the other values of $n$, we substitute again $l$ by $2l$ in the first term, and by $2l+1$ in the second term, to concluded the proof of (3.26).

□

As before, in order to compute the fractional Laplacian of the cosine-like and sine-like Christov functions, we first express (3.19) as a polynomial on $x$ multiplied by a rational function as follows:

**Lemma 3.2.2.** *Let $\mu_n(x)$ be defined as in* (3.5)*. Then,* (3.19) *can be expanded as*

$$\begin{aligned}&(-\Delta)^{\alpha/2}\mu_n(x)\\ &=\begin{cases}\dfrac{\Gamma(1+\alpha)}{(ix+1)^{n+1+\alpha}}\displaystyle\sum_{l=0}^{n}\binom{n}{l}(ix)^{n-l}{}_2F_1(-l,1+\alpha;1;2), & n=0,1,2,\dots,\\ -\dfrac{\Gamma(1+\alpha)}{(-ix+1)^{-n+\alpha}} & \\ \qquad\cdot\displaystyle\sum_{l=0}^{-1-n}\binom{-1-n}{l}(-ix)^{-1-n-l}{}_2F_1(-l,1+\alpha;1;2), & n=-1,-2,-3,\dots.\end{cases}\end{aligned}\tag{3.28}$$

*Proof.* Assume $n$ is a nonnegative number. The proof is almost identical to that of Proposition 3.2.1. Indeed, starting from the second last line in (3.19) and following

the same steps as in (3.23) and (3.24),

$$\begin{aligned}(-\Delta)^{\alpha/2}\mu_n(x) &= \frac{2^n\Gamma(1+\alpha)}{(ix+1)^{n+1+\alpha}}\sum_{k=0}^{n}\sum_{l=0}^{k}\frac{1}{2^k}\binom{n}{k}\binom{k}{l}\binom{-1-\alpha}{n-k}(ix)^l \\ &= \frac{\Gamma(1+\alpha)}{(ix+1)^{n+1+\alpha}}\sum_{l=0}^{n}\binom{n}{l}(ix)^{n-l}\sum_{k=0}^{l}\binom{l}{k}\binom{-1-\alpha}{k}2^k,\end{aligned}$$

which is (3.28). The case with $n$ a negative integer follows from (3.21).

□

*Remark.* As we did for $(-\Delta)^{\alpha/2}\lambda_n(x)$, if we only consider $l=0$ in (3.28), we get the asymptotic behavior of $(-\Delta)^{\alpha/2}\mu_n(x)$, as $x\to\pm\infty$:

$$(-\Delta)^{\alpha/2}\mu_n(x) \sim \begin{cases}\dfrac{\Gamma(1+\alpha)(ix)^n}{(ix+1)^{n+1+\alpha}} \sim \dfrac{\Gamma(1+\alpha)}{(ix)^{1+\alpha}}, & n=0,1,2,\dots, \\ -\dfrac{\Gamma(1+\alpha)(-ix)^{-1-n}}{(-ix+1)^{-n+\alpha}} \sim -\dfrac{\Gamma(1+\alpha)}{(-ix)^{1+\alpha}}, & n=-1,-2,-3,\dots;\end{cases}$$

whereas

$$\mu_n(x) = \frac{1}{2}\left[\left(1-\frac{2}{ix+1}\right)^n - \left(1-\frac{2}{ix+1}\right)^{n+1}\right] \sim \frac{1}{ix+1} \sim \frac{1}{ix}, \quad x\to\pm\infty.$$

Therefore, as expected, the operator $(-\Delta)^{\alpha/2}$ applied on $\mu_n(x)$ introduces an extra decay of $|x|^{-\alpha}$.

As a consequence of Lemma 3.2.2, we have the following result.

**Proposition 3.2.3.** *The fractional Laplacian of* (3.8) *and* (3.9) *is given respectively by*

$$\begin{aligned}(-\Delta)^{\alpha/2}\mathrm{CC}_{2n}(x) = {} & \frac{2n\Gamma(1+\alpha)}{(1+x^2)^{(n+1+\alpha)/2}}\sin\left((n+1+\alpha)\operatorname{arccot}(x)-\frac{\pi\alpha}{2}\right) \\ & \sum_{l=0}^{\lfloor\frac{n}{2}\rfloor}(-1)^l\binom{n}{2l}x^{n-2l}{}_2F_1(-2l,1+\alpha;2;1) \\ & -\frac{2n\Gamma(1+\alpha)}{(1+x^2)^{(n+1+\alpha)/2}}\cos\left((n+1+\alpha)\operatorname{arccot}(x)-\frac{\pi\alpha}{2}\right) \\ & \sum_{l=0}^{\lfloor\frac{n-1}{2}\rfloor}(-1)^l\binom{n}{2l+1}x^{n-2l-1}{}_2F_1(-2l-1,1+\alpha;2;1), \quad (3.29)\end{aligned}$$

*and*

$$(-\Delta)^{\alpha/2}\,\mathrm{SC}_{2n+1}(x) = \frac{2n\Gamma(1+\alpha)}{(1+x^2)^{(n+1+\alpha)/2}}\cos\left((n+1+\alpha)\,\mathrm{arccot}(x)-\frac{\pi\alpha}{2}\right)$$
$$\sum_{l=0}^{\lfloor\frac{n}{2}\rfloor}(-1)^l\binom{n}{2l}x^{n-2l}{}_2F_1(-2l,1+\alpha;2;1)$$
$$+\frac{2n\Gamma(1+\alpha)}{(1+x^2)^{(n+1+\alpha)/2}}\sin\left((n+1+\alpha)\,\mathrm{arccot}(x)-\frac{\pi\alpha}{2}\right)$$
$$\sum_{l=0}^{\lfloor\frac{n-1}{2}\rfloor}(-1)^l\binom{n}{2l+1}x^{n-2l-1}{}_2F_1(-2l-1,1+\alpha;2;1). \quad (3.30)$$

*Proof.* The proof is very similar to that of Proposition 3.2.2, so we omit some details. We first observe that, from (3.8), (3.9) and (3.21) (see also [7]), for $n=0,1,2,\ldots$,

$$\mathrm{CC}_{2n}(x)=\Re(\mu_n(x)),\qquad \mathrm{SC}_{2n+1}(x)=-\Im(\mu_{n+1}(x)),$$

hence,

$$\mathrm{CC}_{2n}(\cot(s))=\frac{\cos(2ns)-\cos(2(n+1)s)}{2},$$
$$\mathrm{SC}_{2n+1}(\cot(s))=\frac{\sin(2(n+1)s)-\sin(2ns)}{2}.$$

We treat first the case where $n$ is a nonnegative integer. Then, we express all the complex numbers appearing in (3.28) in their polar form, to obtain:

$$(-\Delta)^{\alpha/2}\mu_n(x)=\frac{\Gamma(1+\alpha)}{(1+x^2)^{(n+1+\alpha)/2}}\sum_{l=0}^{n}\binom{n}{l}x^{n-l}$$
$$e^{i((n+1+\alpha)\,\mathrm{arccot}(x)-\pi(l+1+\alpha)/2)}{}_2F_1(-l,1+\alpha;1;2). \quad (3.31)$$

Now, taking the real part of (3.31) gives

$$(-\Delta)^{\alpha/2}\,\mathrm{CC}_{2n}(x)=\frac{\Gamma(1+\alpha)}{(1+x^2)^{(n+1+\alpha)/2}}\sum_{l=0}^{n}\binom{n}{l}x^{n-l}$$
$$\sin\left((n+1+\alpha)\cot(x)-\frac{\pi\alpha}{2}-\frac{\pi l}{2}\right){}_2F_1(-l,1+\alpha;2;1)$$
$$=\frac{2n\Gamma(1+\alpha)}{(1+x^2)^{(n+1+\alpha)/2}}\sin\left((n+1+\alpha)\,\mathrm{arccot}(x)-\frac{\pi\alpha}{2}\right)$$
$$\sum_{l=0}^{n}\cos\left(\frac{\pi l}{2}\right)\binom{n}{l}x^{n-l}{}_2F_1(-l,1+\alpha;2;1)$$
$$-\frac{2n\Gamma(1+\alpha)}{(1+x^2)^{(n+1+\alpha)/2}}\cos\left((n+1+\alpha)\,\mathrm{arccot}(x)-\frac{\pi\alpha}{2}\right)$$
$$\sum_{l=0}^{n}\sin\left(\frac{\pi l}{2}\right)\binom{n}{l}x^{n-l}{}_2F_1(-l,1+\alpha;2;1).$$

Substituting $l$ by $2l$ in the first term, and by $2l+1$ in the second term of the last expression, we get (3.29).

Likewise, taking the imaginary part of (3.27) and changing the sign, we have

$$\begin{aligned}(-\Delta)^{\alpha/2}\,\mathrm{SC}_{2n+1}(x) &= \frac{\Gamma(1+\alpha)}{(1+x^2)^{(n+1+\alpha)/2}}\sum_{l=0}^{n}\binom{n}{l}x^{n-l}\\ &\qquad \cos\left((n+1+\alpha)\cot(x)-\frac{\pi\alpha}{2}-\frac{\pi l}{2}\right){}_2F_1(-l,1+\alpha;2;1)\\ &= \frac{2n\Gamma(1+\alpha)}{(1+x^2)^{(n+1+\alpha)/2}}\cos\left((n+1+\alpha)\,\mathrm{arccot}(x)-\frac{\pi\alpha}{2}\right)\\ &\quad \sum_{l=0}^{n}\cos\left(\frac{\pi l}{2}\right)\binom{n}{l}x^{n-l}{}_2F_1(-l,1+\alpha;2;1)\\ &\quad +\frac{2n\Gamma(1+\alpha)}{(1+x^2)^{(n+1+\alpha)/2}}\sin\left((n+1+\alpha)\,\mathrm{arccot}(x)-\frac{\pi\alpha}{2}\right)\\ &\quad \sum_{l=0}^{n}\sin\left(\frac{\pi l}{2}\right)\binom{n}{l}x^{n-l}{}_2F_1(-l,1+\alpha;2;1).\end{aligned}$$

Finally, substituting $l$ by $2l$ in the first term, and by $2l+1$ in the second term of the last expression, we get (3.30). □

### 3.2.1 Cases with $\alpha \in \{0,1,2\}$

Since we are working with $\alpha \in (0,2)$, it is interesting to see what happens if we evaluate (3.4), (3.19) and their cosine-like and sine-like expressions at the ends and at the middle point of the interval. For that purpose, we first express $\lambda_n(x)$ in (3.2) as

$$\lambda_n(x) = \left(\frac{i\,\mathrm{sgn}(n)x-1}{i\,\mathrm{sgn}(n)x+1}\right)^{|n|};$$

hence,

$$\begin{aligned}\lambda_n'(x) &= \frac{2in}{(i\,\mathrm{sgn}(n)x+1)^2}\left(\frac{i\,\mathrm{sgn}(n)x-1}{i\,\mathrm{sgn}(n)x+1}\right)^{|n|-1},\\ \lambda_n''(x) &= \frac{4inx-4n^2}{(i\,\mathrm{sgn}(n)x+1)^4}\left(\frac{i\,\mathrm{sgn}(n)x-1}{i\,\mathrm{sgn}(n)x+1}\right)^{|n|-2}.\end{aligned}$$

We also need Newton's binomial formula and its derivative:

$$(1+z)^{n-1}=\sum_{k=0}^{n-1}\binom{n-1}{k}z^k \Longrightarrow (n-1)(1+z)^{n-2}=\sum_{k=1}^{n-1}k\binom{n-1}{k}z^{k-1}.$$

Let us consider first the case with $\alpha = 0$, $n \in \mathbb{Z}\backslash\{0\}$:

$$\begin{aligned}\lim_{\alpha\to 0^+}(-\Delta)^{\alpha/2}\lambda_n(x) &= -\frac{2|n|\Gamma(1)}{i\,\mathrm{sgn}(n)x+1}{}_2F_1\left(1-|n|,1;2;\frac{2}{i\,\mathrm{sgn}(n)x+1}\right)\\ &= \sum_{k=0}^{|n|-1}\binom{|n|}{k+1}\left(\frac{-2}{i\,\mathrm{sgn}(n)x+1}\right)^{k+1}\\ &= \sum_{k=1}^{|n|}\binom{|n|}{k}\left(\frac{-2}{i\,\mathrm{sgn}(n)x+1}\right)^{k}\\ &= \left(1-\frac{2}{i\,\mathrm{sgn}(n)x+1}\right)^{|n|}-1\\ &= \lambda_n(x)-1.\end{aligned}$$

Therefore, for $n = 0, 1, 2, \ldots$,

$$\begin{aligned}\lim_{\alpha\to 0^+}(-\Delta)^{\alpha/2}\,\mathrm{CH}_{2n}(x) &= \mathrm{CH}_{2n}(x)-1,\\ \lim_{\alpha\to 0^+}(-\Delta)^{\alpha/2}\,\mathrm{SH}_{2n+1}(x) &= \mathrm{SH}_{2n+1}(x).\end{aligned}$$

Observe that, as with $\lambda_0(x) = 1$ in (3.1), this limit is singular for $\lambda_n(x)$ and $\mathrm{CH}_{2n}$, since their respective Fourier transforms are not classical functions. On the other hand, the limit for $\mathrm{SH}_{2n+1}(x)$ is continuous, because the function belongs to $\mathcal{L}^2(\mathbb{R})$, and, hence, its Fourier transform is well defined. Therefore, if we consider instead of $\lambda_n(x)$ and $\mathrm{CH}_{2n}(x)$, $\lambda_n(x) - 1$ and $\mathrm{CH}_{2n} - 1$, respectively, the functions belong now to $\mathcal{L}^2(\mathbb{R})$, and

$$\begin{aligned}\lim_{\alpha\to 0^+}(-\Delta)^{\alpha}(\lambda_n(x)-1) &= \lambda_n(x)-1\\ &= (-\Delta)^0(\lambda_n(x)-1),\\ \lim_{\alpha\to 0^+}(-\Delta)^{\alpha}(\mathrm{CH}_{2n}(x)-1) &= \mathrm{CH}_{2n}(x)-1\\ &= (-\Delta)^0(\mathrm{CH}_{2n}(x)-1),\end{aligned}$$

i.e., the limit is now continuous. Likewise, the limit is also continuous for $\mu_n(x)$ (and, hence, for $\mathrm{CC}_{2n}(x)$ and $\mathrm{SC}_{2n+1}(x)$), because they are in $\mathcal{L}^2(\mathbb{R})$. Indeed, from (3.20),

$$\begin{aligned}\lim_{\alpha\to 0^+}(-\Delta)^{\alpha/2}\mu_n(x) &= \lim_{\alpha\to 0^+}\frac{(-\Delta)^{\alpha/2}\lambda_n(x)-(-\Delta)^{\alpha/2}\lambda_{n+1}(x)}{2}\\ &= \frac{\lambda_n(x)-\lambda_{n+1}(x)}{2}\\ &= (-\Delta)^0\mu_n(x).\end{aligned}$$

Unlike the previous case, the case with $\alpha = 2$ poses no problems:

$$\begin{aligned}
\lim_{\alpha\to 2}(-\Delta)^{\alpha/2}\lambda_n(x) &= -\frac{2|n|\Gamma(3)}{(i\operatorname{sgn}(n)x+1)^3}{}_2F_1\left(1-|n|,3;2;\frac{2}{i\operatorname{sgn}(n)x+1}\right)\\
&= -\frac{2|n|}{(i\operatorname{sgn}(n)x+1)^3}\sum_{k=0}^{|n|-1}\binom{|n|-1}{k}(k+2)\left(-\frac{2}{i\operatorname{sgn}(n)x+1}\right)^k\\
&= \frac{4|n|(|n|-1)}{(i\operatorname{sgn}(n)x+1)^4}\left(1-\frac{2}{i\operatorname{sgn}(n)x+1}\right)^{|n|-2}\\
&\quad -\frac{4|n|}{(i\operatorname{sgn}(n)x+1)^3}\left(1-\frac{2}{i\operatorname{sgn}(n)x+1}\right)^{|n|-1}\\
&= -\lambda_n''(x),
\end{aligned}$$

i.e., we recover $(-\Delta)^1\lambda_n(x)$. This is also true for all $\mu_n(x)$, $\mathrm{CH}_{2n}(x)$, $\mathrm{SH}_{2n+1}(x)$, $\mathrm{CC}_{2n}(x)$ and $\mathrm{SC}_{2n+1}(x)$. Finally, when $\alpha = 1$,

$$\begin{aligned}
\lim_{\alpha\to 1}(-\Delta)^{\alpha/2}\lambda_n(x) &= -\frac{2|n|\Gamma(2)}{(i\operatorname{sgn}(n)x+1)^2}{}_2F_1\left(1-|n|,2;2;\frac{2}{i\operatorname{sgn}(n)x+1}\right)\\
&= -\frac{2|n|}{(i\operatorname{sgn}(n)x+1)^2}\sum_{k=0}^{|n|-1}\binom{|n|-1}{k}\left(-\frac{2}{i\operatorname{sgn}(n)x+1}\right)^k\\
&= -\frac{2|n|}{(i\operatorname{sgn}(n)x+1)^2}\left(1-\frac{2}{i\operatorname{sgn}(n)x+1}\right)^{|n|-1}\\
&= i\operatorname{sgn}(n)\lambda_n'(x).
\end{aligned}$$

Note that, if we express the last equality in terms of $s$, we obtain $2|n|\sin^2(s)e^{i2ns}$, which is precisely the expression for $(-\Delta)^{1/2}\lambda_n(\cot(s))$ obtained in Chapter 2 using a completely different approach. Consequently, for $n = 0, 1, 2, \ldots$, from (3.6) and (3.7), respectively,

$$\begin{aligned}
(-\Delta)^{1/2}\,\mathrm{CH}_{2n}(x) &= \frac{i\operatorname{sgn}(n)\lambda_n'(x) + i\operatorname{sgn}(-n)\lambda_{-n}'(x)}{2}\\
&= \begin{cases} 0, & n = 0,\\ -\mathrm{SH}_{2n-1}'(x), & n\in\mathbb{N}, \end{cases}\\
(-\Delta)^{1/2}\,\mathrm{SH}_{2n+1}(x) &= \frac{i\operatorname{sgn}(n+1)\lambda_{n+1}'(x) - i\operatorname{sgn}(-n-1)\lambda_{-n-1}'(x)}{2i}\\
&= \mathrm{CH}_{2n+2}'(x).
\end{aligned}$$

Similarly, in the case of $\mu_n(x)$,

$$\begin{aligned}
(-\Delta)^{1/2}\mu_n(x) &= \frac{i\operatorname{sgn}(n)\lambda_n'(x) - i\operatorname{sgn}(n+1)\lambda_{n+1}'(x)}{2}\\
&= \begin{cases} i\mu_n'(x), & n = 0, 1, 2, \ldots,\\ -i\mu_n'(x), & n = -1, -2, -3, \ldots, \end{cases}
\end{aligned}$$

or, equivalently, $(-\Delta)^{1/2}\mu_n(x) = i\,\mathrm{sgn}(n+1/2)\mu_n'(x)$. Therefore, for $n = 0, 1, 2, \ldots$, from (3.8) and (3.9), respectively,

$$\begin{aligned}(-\Delta)^{1/2}\,\mathrm{CC}_{2n}(x) &= \frac{i\,\mathrm{sgn}(n+1/2)\mu_n'(x) - i\,\mathrm{sgn}(-n-1/2)\mu_{-n-1}'(x)}{2}\\ &= \mathrm{SC}_{2n+1}'(x),\\ (-\Delta)^{1/2}\,\mathrm{SC}_{2n+1}(x) &= -\frac{i\,\mathrm{sgn}(n+1/2)\mu_n'(x) + i\,\mathrm{sgn}(-n-1/2)\mu_{-n-1}'(x)}{2i}\\ &= -\,\mathrm{CC}_{2n}'(x).\end{aligned}$$

Observe that all the equalities for $\alpha = 1$ can be derived from the fact that $\lambda_n(x)$ and $\mu_n(x)$ are eigenfunctions of the Hilbert transform (see for instance [64, 65]).

## 3.3 Numerical implementation of (3.4)

In this Section, we will focus on the numerical implementation of (3.4), because the other functions considered in this chapter can be expressed in terms of $\lambda_n(x)$. Moreover, we will reduced ourselves to the case with $n \in \mathbb{N}$, because the case with nonpositive $n$ is trivially reduced to the former. All the experiments that follow have been carried out in an HP ZBook 15 G3 with processor Intel(R) Core(TM) i7-6700HQ CPU @ 2.60GHz, graphic card NVIDIA Quadro M1000M, and 16384MB of RAM, under Windows 7 Enterprise with Service Pack 1. We have mainly worked with MATLAB R2019b [8], but have also used MATHEMATICA 11.3 [105] in a few examples.

Let us start by discussing the drawbacks of attempting to implement (3.4) by using implementations of ${}_2F_1$ in commercial mathematical software. Even if commands that approximate numerically ${}_2F_1$ functions are available in, among others, MATLAB and MATHEMATICA, it seems that there is still no satisfactory implementation of ${}_2F_1$ for all its possible values. For example, in our case, there seem to be cancellation errors, due to the fact that we are adding and subtracting alternatively very large numbers. Recall that the infinite series in (3.10) converges in general for $|z| < 1$, which in our case means $|z| = |2/(ix+1)| < 1$ or $|x| > \sqrt{3}$, even if we are interested in evaluating it at any $x \in \mathbb{R}$. On the other hand, since the first parameter of ${}_2F_1$ in (3.4), $1-n$, is negative or zero, it implies that the numerical approximation of (3.4) is reduced to computing a finite sum, which is always convergent, so, in principle, we should expect no serious accuracy issues from a commercial implementation. However, if we consider for instance $x = 0$, for which $|2/(ix+1)|$ is largest (and, hence, the worst case), it is easy to check that both MATLAB and MATHEMATICA return wrong results for not too large values of $n$. Take for example $n = 400$ and $\alpha = 1.5$; then, the MATHEMATICA command

```
Hypergeometric2F1[-399, 2.5, 2, 2]
```

gives $6.2771 \cdot 10^{57}$, whereas the MATLAB command

```
hypergeom([-399, 2.5], 2, 2)
```

yields $4.9217 \cdot 10^{18}$. This is due to the fact that the signs of $(1-n)_k$ alternate, whereas $(1+\alpha)_k$ is always positive; hence, we are summing large quantities with alternating signs, for which a 64-bit floating point representation is clearly insufficient to store all the required digits, giving rise to severe rounding errors. Nevertheless, it is possible to obtain the correct results with both programs by just introducing minor modifications. More precisely, both

```
N[Hypergeometric2F1[-399, 5/2, 2, 2]]
```

and

```
double(hypergeom([-399, str2sym('5/2')], 2, 2))
```

return the correct result, ${}_2F_1(-399, 2.5; 2; 2) = -21.2769\ldots$. In fact, if we omit respectively the commands `N` and `double`, we get a fraction with very large nominator and denominator. Observe that, in MATHEMATICA, `5/2` is treated natively in a different way as `2.5`, which is stored using a floating point representation; whereas, in MATLAB, to obtain a similar effect, the Symbolic Math Toolbox needs to be installed. Then,

```
str2sym('5/2')
```

transforms the string `'5/2'` into a symbolic object that stores the fraction $5/2$.

In the rest of the section, for the sake of simplicity, we will work exclusively with MATLAB. The previous arguments suggest working with fractions whenever possible. However, if some of the parameters of ${}_2F_1$ are irrational, approximating them adequately by fractions might involve large numerators and denominators, which does not seem ideal, either. Instead, we have found more advisable to work with a larger number of digits, which can be done by means of the function `vpa` included in the Symbolic Math Toolbox. The name of this command comes from the initials of "variable-precision arithmetic", i.e., arbitrary-precision arithmetic, and evaluates by default its argument to at least 32 digits, even if another number of digits can be specified as a second argument of `vpa`; e.g.,

```
vpa(str2sym('pi'), 500)
```

returns 500 exact digits of $\pi$. Furthermore, it is possible to change globally the default number of digits to $d$ by means of `digits(d)`. Take for instance $n = 400$, $\alpha = \sqrt{3}$, then

```
hypergeom([-399, 1 + sqrt(3)], 2, 2)
```

returns $-1.4831 \cdot 10^{18}$; whereas `digits(24)`; followed by

```
hypergeom([-399, vpa(str2sym('1 + sqrt(3)'))], 2, 2)
```

returns the exact value of the first 24 digits of ${}_2F_1(-399, 1+\sqrt{3}; 2; 2)$, namely

$$-84.174204314895374080 4326$$

Therefore, in our opinion, the use of arbitrary precision is the easiest and safest way to guarantee that the results returned by ${}_2F_1$ are correct, and, indeed, research is currently been conducted on this area (see for instance [131] and its references,

where ${}_2F_1$ and other hypergeometric functions are computed in arbitrary-precision interval arithmetic).

Coming back to the implementation of (3.4), note that `z` in

```
hypergeom([a,b],c,z)
```

can be a vector, but `a`, `b` and `c` must be scalars. Therefore, if we have a function $u(x)$ represented as say

$$u(x) = \sum_{n=-N}^{N} a_n \lambda_n(x),$$

and want to approximate its fractional Laplacian by using `hypergeom`, we have to invoke this command $N$ times using arbitrary precision, i.e. for $n = 1, \dots, N$, which can be extremely expensive from a computational point of view. For example, executing

```
hypergeom([1-n, vpa(str2sym('1 + sqrt(3)'))], 2, 2)
```

for $n = 1, 2, \dots, 40$, using 24 digits, requires 24.31 seconds, even if we are considering just one single value of $z$. To avoid this, we compute (3.4) by using directly a series representation, but in such a way that the evaluations for different $n$ are done simultaneously (and as much independently from $\alpha$ as possible), which is much more efficient. More precisely, we express (3.4) in the following form and evaluate it at $M$ different spatial nodes $x_0, \dots, x_{M-1}$:

$$(-\Delta)^{\alpha/2}\lambda_n(x_j) = -\Gamma(1+\alpha)(ix+1)^{-\alpha} \cdot \sum_{k=1}^{n} \binom{n}{k} \binom{-1-\alpha}{k-1} \left(\frac{2}{ix_j+1}\right)^k, \quad n \in \{1, \dots, N\}. \tag{3.32}$$

After converting $\alpha$ and $\pi$ to arbitrary precision, we generate the spatial nodes and store them in one column vector $\mathbf{x} = (x_0, \dots, x_{M-1})^T$. Then, we create a matrix $\mathbf{A} \in \mathcal{M}_{M\times N}(\mathbb{C})$, such that its columns are precisely the powers $\{1, \dots, N\}$ of the entries of

$$\left(\frac{2}{ix_0+1}, \dots, \frac{2}{ix_{M-1}+1}\right)^T,$$

i.e.,

$$\mathbf{A} = \begin{pmatrix} \left(\frac{2}{ix_0+1}\right)^1 & \cdots & \left(\frac{2}{ix_0+1}\right)^N \\ \vdots & \ddots & \vdots \\ \left(\frac{2}{ix_{M-1}+1}\right)^1 & \cdots & \left(\frac{2}{ix_{M-1}+1}\right)^N \end{pmatrix}.$$

This can be done in parallel by just typing

```
A = (2 ./ (1i * x + 1)) .^ (1:N);
```

Afterward, we create a matrix $\mathbf{B} \in \mathcal{M}_{N\times N}(\mathbb{C})$ that stores the binomial coefficients computed using arbitrary precision:

$$\mathbf{B} = \begin{pmatrix} \binom{1}{1} & \binom{2}{1} & \binom{3}{1} & \dots & \binom{N}{1} \\ 0 & \binom{2}{2} & \binom{3}{2} & \dots & \binom{N}{2} \\ 0 & 0 & \binom{3}{3} & \dots & \binom{N}{3} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & \binom{N}{N} \end{pmatrix},$$

which can be done column-wise by using

$$\binom{n+1}{k+1} = \binom{n}{k+1} + \binom{n}{k}.$$

However, if we apply recursively this property, we get

$$\binom{n+1}{k+1} = \binom{n}{k} + \binom{n}{k+1} = \binom{n}{k} + \binom{n-1}{k} + \binom{n-1}{k+1} = \dots = \sum_{j=0}^{n} \binom{j}{k},$$

which allows to generate $\mathbf{B}$ row-wise as well.

This alternative implementation appears to be faster and is the one we have adopted. More precisely, after initializing $\mathbf{B}$ with

```
B = vpa(zeros(N, N));
```

we store the first row, which is explicitly known, i.e.,

```
B(1, :)  = 1:N;
```

Then, each remaining row is generated recursively by computing the cumulative sum of the entries of the previous row, i.e.,

```
for n = 2:N, B(n, n:N) = cumsum(B(n-1, n-1:N-1)); end
```

At this point, it is important to underline that both $\mathbf{A}$ and $\mathbf{B}$ are completely independent from $\alpha$, so they can be generated once and then be reused to evaluate (3.32) at different values of $\alpha$. Furthermore, the parts of (3.32) that depend actually on $\alpha$ (which we store in an arbitrary-precision variable `a`) can be stored in just two vectors $\mathbf{c} \in \mathcal{M}_{M\times 1}(\mathbb{C})$ and $\mathbf{d} \in \mathcal{M}_{1\times N}(\mathbb{C})$; the former is a column vector containing the entries of $i\mathbf{x}+1$ raised to the power $-\alpha$, i.e.,

```
c = (1i * x + 1) .^ (-a);
```

whereas the latter is a row vector formed by the values

$$-\Gamma(1+\alpha)\binom{-1-\alpha}{k-1}, \quad k = 1, \dots, N.$$

Moreover, we observe that the entries of $\mathbf{d}$ can be generated by obtaining the cumulative product of the following quantities:

$$\left\{-\Gamma(1+\alpha), \frac{-1-\alpha}{1}, \frac{-2-\alpha}{2}, \ldots, \frac{-(N+1)-\alpha}{N-1}\right\},$$

i.e.,

```
d = cumprod([-gamma(1 + a), -1 - a ./ (1:N-1)]);
```

At this point, combining all the previously defined variables elements, we generate a matrix $\mathbf{M}_\alpha \in \mathcal{M}_{M\times N}(\mathbb{C})$, such that its columns correspond to (3.32) evaluated at $n = 1, \ldots, N$, respectively:

$$\mathbf{M}_\alpha = \operatorname{diag}(\mathbf{c})\mathbf{A}\operatorname{diag}(\mathbf{d})\mathbf{B},$$

where $\operatorname{diag}(\mathbf{c}) \in \mathcal{M}_{M\times M}(\mathbb{C})$ and $\operatorname{diag}(\mathbf{d}) \in \mathcal{M}_{N\times N}(\mathbb{C})$ are the diagonal matrices whose diagonal entries are $\mathbf{c}$ and $\mathbf{d}$ respectively; in MATLAB,

```
Ma = diag(c) * A * diag(d) * B;
```

In this regard, let us mention that we have found that this equivalent expression executes a bit faster:

$$\mathbf{M}_\alpha = (\mathbf{A} \circ (\mathbf{c} \cdot \mathbf{d}))\mathbf{B},$$

where $\circ$ denotes the Hadamard product, i.e., the entrywise product between matrices or equal size. In that case,

```
Ma = (A .* (c * d)) * B;
```

Finally, the resulting matrix can be cast to the 64-bit floating point format by means of

```
Ma = double(Ma);
```

Observe that, if $\mathbf{M}_\alpha$ is very large, it might happen that `double` cannot be applied to a whole arbitrary-precision matrix, because the capacity of MATLAB is exceeded; in that case, the conversion has to be done row-wise or column-wise, although the global impact is small.

In the following subsection, we will perform the numerical experiments; special attention will be given to the number of digits necessary to compute $\mathbf{M}_\alpha$ with satisfactory accuracy.

### 3.3.1 Numerical experiments

The procedure to generate $\mathbf{M}_\alpha \in \mathcal{M}_{M\times N}(\mathbb{C})$ that we have just explained is by definition *exact*, provided that a large enough number of digits is chosen. In our numerical experiments, we have observed that the number of necessary digits is virtually independent from the choice of $\alpha \in (0, 2)$; in what follows, we have used systematically the irrational value $\alpha = \sqrt{3}$. We have first considered only the worst case $x = 0$,

and have thus approximated $(-\Delta)^{\sqrt{3}/2}\lambda_N(0)$, i.e., have computed the last element of the row vector $\mathbf{M}_{\sqrt{3}} \in \mathcal{M}_{1\times N}(\mathbb{C})$ using arbitrary precision with an increasing number of digits, until satisfactory convergence is achieved. Let ${m_N}^{(k)}$ denote the $k$-digit approximation of $(-\Delta)^{\sqrt{3}/2}\lambda_N(0)$; then, we take as stopping criterion that

$$\left| \frac{m_N^{(k)} - m_N^{(k+1)}}{m_N^{(k+1)}} \right| < \varepsilon, \tag{3.33}$$

where $\varepsilon = 2^{-52}$, eps in MATLAB, is the so-called machine epsilon in 64-bit floating point arithmetic. Note that the use of the relative discrepancy between ${m_N}^{(k)}$ and ${m_N}^{(k+1)}$ is justified because $|(-\Delta)^{\sqrt{3}/2}\lambda_N(0)|$ quickly grows with $N$ (we have found experimentally that $|(-\Delta)^{\sqrt{3}/2}\lambda_N(0)| \approx 3.322N^{\sqrt{3}}$), and we are interested in choosing the adequate number of global digits (before and after the decimal point), not of correct decimals. Remark that, in the numerical experiments, we have found that, in general, the number of digits required does not diminish with $N$, even if, very occasionally, that quantity can be slightly smaller for $N+1$ than for $N$. Hence, in order to determine the number of digits for a given $N+1$, it is completely safe to start generating ${m_{N+1}}^{(k)}$ with the same number of digits $k$ necessary to generate ${m_N}^{(k)}$ satisfying (3.33), and then increase $k$ by one unity, until the convergence condition (3.33) is fulfilled. In Figure 3.2, we have plotted the number of digits with respect to $N$; note that the graph is roughly a straight line with slope slightly less than $1/2$; in fact, the least squares regression line is $\hat{y} = 0.47598x + 6.6938$. On the other hand,

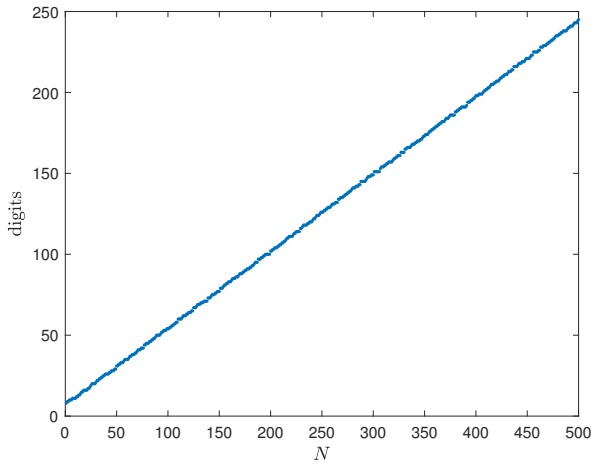


FIGURE 3.2: Number of digits as a function of $N$ necessary to approximate $\mathbf{M}_\alpha \in \mathcal{M}_{1\times N}(\mathbb{C})$ with accuracy (3.33)

for a given number of columns $N$, we have also considered $N$ nodes $\{x_0, \ldots, x_{N-1}\}$ and approximated the whole matrix $\mathbf{M}_{\sqrt{3}} \in \mathcal{M}_{N\times N}(\mathbb{C})$. In order to compare the results with those in Chapter 2, we have taken equally-spaced non-end nodes $s_j$, which determine the choice of $x_j$:

$$s_j = \frac{\pi(2j+1)}{2N} \Longrightarrow x_j = \cot\left(\frac{\pi(2j+1)}{2N}\right), \quad 0 \leq j \leq N-1. \tag{3.34}$$

Let $\mathbf{M}_{\sqrt{3}}^{(k)} \in \mathcal{M}_{N\times N}(\mathbb{C})$ denote the approximation of $(-\Delta)^{(\sqrt{3}/2)}\lambda_n(x_j)$, for $n \in \{1,\dots,N\}$, $j \in \{0,\dots,N-1\}$, using arbitrary precision arithmetic with $k$ digits. Bearing in mind that $x_{N-1-j} = -x_j$, we have

$$(-\Delta)^{(\sqrt{3}/2)}\lambda_n(x_{N-1-j}) = (-\Delta)^{(\sqrt{3}/2)}\lambda_n(-x_j) = \overline{(-\Delta)^{(\sqrt{3}/2)}\lambda_n(x_j)},$$

so the last $\lfloor N/2 \rfloor$ rows of $\mathbf{M}_{\sqrt{3}}^{(k)}$ can be generated from its first $\lfloor N/2 \rfloor$ rows, which reduces the global computational cost. Observe that, from (3.22), it follows that $\lim_{x\to\pm\infty}(-\Delta)^{\sqrt{3}/2}|\lambda_n(x)| = 0$ in (3.4), for all $n$, so, in general, we can expect very large differences in the orders of magnitude of the different entries of $\mathbf{M}_\alpha$. Therefore, denoting $\mathbf{M}_\alpha{}^{(k)} = [m_{ij}{}^{(k)}]$ and $\mathbf{M}_\alpha{}^{(k+1)} = [m_{ij}{}^{(k+1)}]$, we take now as stopping criterion that

$$\max_{ij}\left(\min\left\{\left|m_{ij}^{(k)} - m_{ij}^{(k+1)}\right|, \left|\frac{m_{ij}^{(k)} - m_{ij}^{(k+1)}}{m_{ij}^{(k+1)}}\right|\right\}\right) < \varepsilon, \tag{3.35}$$

where, again, $\varepsilon = 2^{-52}$. We have found this criterion to be very adequate, because, when $m_{ij}{}^{(k)}$ and $m_{ij}{}^{(k+1)}$ are infinitesimal, the absolute discrepancy is enough to establish convergence, whereas, for larger values, it is preferable to consider the relative discrepancy.

Similarly as in the previous example, in order to obtain the number of digits for a given $N+1$, we start generating the matrix $\mathbf{M}_{\sqrt{3}}{}^{(k)} \in \mathcal{M}_{(N+1)\times(N+1)}(\mathbb{C})$ with the number of digits $k$ necessary to generate $\mathbf{M}_{\sqrt{3}}{}^{(k)} \in \mathcal{M}_{N\times N}(\mathbb{C})$ satisfying (3.35), and then increase $k$ by one unity, until (3.35) is fulfilled. On the left-hand side of Figure 3.3, we have plotted the number of digits with respect to $N$; the graph is very similar (but not identical) to that in Figure 3.2, the least squares regression line being now $\hat{y} = 0.47777x + 7.3617$; and we have found very similar results for other values of $\alpha$. Obviously, if we consider another nodal distribution such that every node $x_j$ satisfies $2/|ix_j + 1| \ll 1$, the number of digits will be smaller.

We have also compared the approximation of the $N^2$ values $(-\Delta)^{(\sqrt{3}/2)}\lambda_n(x_j)$ via the matrices just generated, which we denote $\mathbf{M}_\alpha^{vpa} \in \mathcal{M}_{N\times N}(\mathbb{C})$, with those given by the method explained in Chapter 2, which we denote $\mathbf{M}_\alpha^{old} \in \mathcal{M}_{N\times N}(\mathbb{C})$; remark that we have adopted the method in Chapter 2 with only minor modification to make it produce matrices of the required size, using $l_{lim} = 1000$, for all the values of $N$. On the right-hand side of Figure 3.3, we have plotted the discrepancy $\mathrm{d}(\mathbf{M}_\alpha^{vpa}, \mathbf{M}_\alpha^{old})$ between both (largely unrelated) techniques, using a formula similar to (3.35):

$$\mathrm{d}(\mathbf{M}_\alpha^1, \mathbf{M}_\alpha^2) \equiv \max_{ij}\left(\min\left\{\left|m_{ij}^1 - m_{ij}^2\right|, \left|\frac{m_{ij}^1 - m_{ij}^2}{m_{ij}^1}\right|\right\}\right); \tag{3.36}$$

such discrepancy is of the order of $10^{-11}$ for the first 500 values of $N$. Note that, although (3.36) is not symmetric, the results provided by $\mathrm{d}(\mathbf{M}_\alpha^{vpa}, \mathbf{M}_\alpha^{old})$ and those

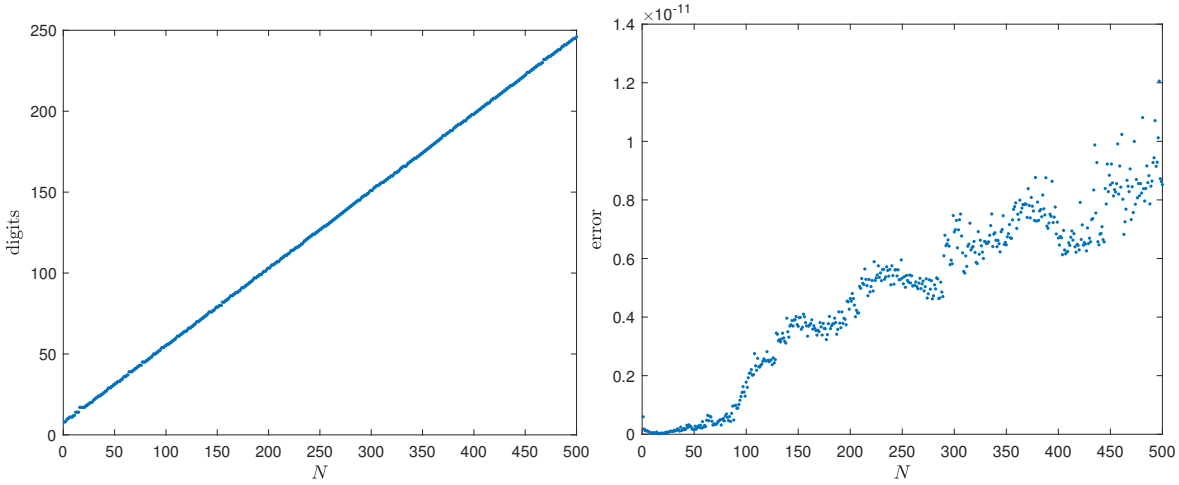


FIGURE 3.3: Left: Number of digits necessary to approximate $\mathbf{M}_\alpha^{vpa} \in \mathcal{M}_{M\times M}(\mathbb{C})$, as a function of $M$, for $\alpha = \sqrt{3}$. Right: Comparison with the results in Chapter 2, using (3.36).

by $\mathrm{d}(\mathbf{M}_\alpha^{old}, \mathbf{M}_\alpha^{vpa})$ show only infinitesimal variations in our case. We have also generated $\mathbf{M}_\alpha^{vpa}$ and $\mathbf{M}_\alpha^{old}$, for $\alpha \in \{0.01, 0.02, \ldots, 1.99\}$, working with 250 digits in the case of $\mathbf{M}_\alpha^{vpa}$; in Figure 3.4, we have plotted $\mathrm{d}(\mathbf{M}_\alpha^{old}, \mathbf{M}_\alpha^{vpa})$ as a function of $\alpha$, which is again of the order of $10^{-11}$. These results confirm the validity of both approaches when approximating the fractional Laplacian.

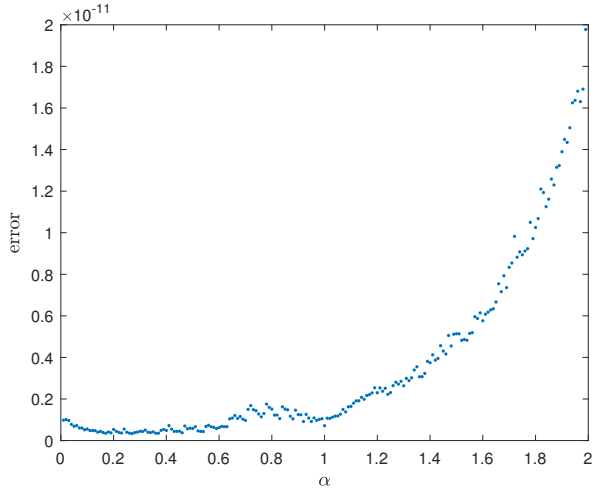


FIGURE 3.4: Comparison with the results in Chapter 2, using (3.36), for $N = 500$ and different values of $\alpha$.

The method that we are describing in this chapter is, in comparison with the method described in Chapter 2, much simpler to implement and *exact* by definition, but, in principle, much more expensive computationally, even if it still largely outperforms the Matlab function `hypergeom`. Recall that `hypergeom` uses transformation formulas like those in [4, Ch. 15.3], which allow it to evaluate ${}_2F_1$ in (3.4), using 64-bit floating point arithmetic, for a much larger range of values of $n$ than it would be possible by simply using the representation (3.12), although it has to be invoked individually for each value of $n$. To give an idea of the time needed for each technique, we have

considered $\mathbf{M}_\alpha \in \mathcal{M}_{N\times N}(\mathbf{C})$, for $N = 330$, i.e., a size for which `hypergeom` produces accurate results without using arbitrary precision. We have generated column-wise $\mathbf{M}_\alpha$ with `hypergeom` using directly (3.4), which we denote $\mathbf{M}_\alpha^{dp}$, where the superscript $dp$ stands for double (i.e., 64-bit) precision, needing 1265.04 sec, whereas $\mathbf{M}_\alpha^{old}$ (based on Chapter 2) and $\mathbf{M}_\alpha^{vpa}$ (using `vpa`) required only 3.44 sec and 54.64 sec, respectively.

To have a more complete picture of the possibilities and limitations of variable-precision arithmetic, we have also tested ADVANPIX [132], the Multiprecision Computing Toolbox for MATLAB. Its main command is `mp`, which stands for multiple precision, and works pretty much in the same way as `vpa`, but, unlike `vpa`, it cannot receive symbolic objects as parameters; e.g., to obtain the multiply precision approximation of $\sqrt{3}$, we just type

```
mp('sqrt(3)')
```

By default, the number of significant digits of `mp` is 34 (quadruple precision); and it can changed globally to $d$ digits by means of

```
mp.Digits(d)
```

or, individually, as a second argument of `mp`. It is important to remark, however, that `vpa` and `mp` do not have exactly the same behavior; e.g, both

```
vpa(str2sym('sqrt(3)'), 2)
```

and

```
mp('sqrt(3)', 2)
```

show 1.7 at the screen, but

```
double(vpa(str2sym('sqrt(3)'), 2)) - sqrt(3)
```

gives $1.1206 \cdot 10^{-10}$, whereas

```
double(mp('sqrt(3)', 2)) - sqrt(3)
```

gives 0.0023. This phenomenon can explained because

```
vpa(str2sym('sqrt(3)'), 2)
```

guarantees at least two digits, even if more digits can be used internally, whereas

```
mp('sqrt(3)', 2)
```

really yields two digits.

Using ADVANPIX, we have generated again $\mathbf{M}_\alpha \in \mathcal{M}_{N\times N}(\mathbb{C})$ for $\alpha = \sqrt{3}$, which we denote $\mathbf{M}_\alpha^{np}$; note that the code is identical to that of $\mathbf{M}_\alpha^{vpa}$, except that we have replaced the three appearances of `vpa` with `mp`, namely `mp('pi')`, `mp('sqrt(3)')` and `mp(zeros(N, N))`. In Figure 3.5, we have plotted the number of digits used to

generate $\mathbf{M}_\alpha^{mp}$, and the least squares regression line is now $\hat{y} = 0.47686x + 14.708$. Therefore, for a given $N$, $\mathbf{M}_\alpha^{mp}$ requires some seven or eight more digits than $\mathbf{M}_\alpha^{vpa}$. On the other hand, the plot of $\mathrm{d}(\mathbf{M}_\alpha^{mp}, \mathbf{M}_\alpha^{old})$ is virtually identical to that on the right-hand side of Figure 3.3, except for infinitesimal variations of the order of $10^{-16}$ for a few values of $N$.

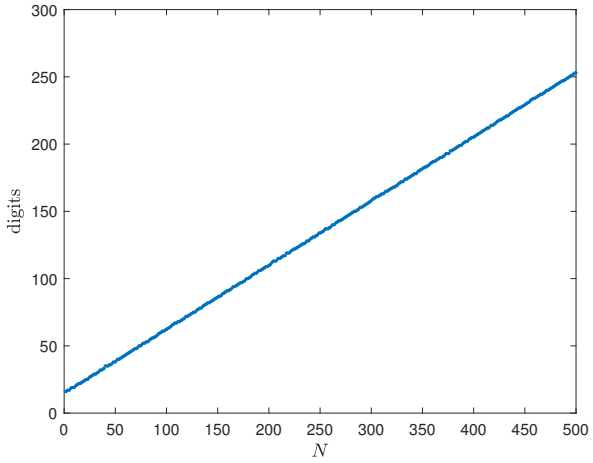


FIGURE 3.5: Number of digits necessary to approximate $\mathbf{M}_\alpha^{mp} \in \mathcal{M}_{M\times M}(\mathbb{C})$, as a function of $M$

In Table 3.1, we show the times required to generate $\mathbf{M}_\alpha \in \mathcal{M}_{M\times M}(\mathbb{C})$ using different techniques and sizes. These times are provided on a purely indicative basis, because they change slightly from execution to execution, and, probably, all of them can be improved by compiling the codes, etc.

| $N$ | $\mathbf{M}_\alpha^{old}$ | $\mathbf{M}_\alpha^{mp}$ | digits | $\mathbf{M}_\alpha^{vpa}$ | digits | $\mathbf{M}_\alpha^{dp}$ |
|---|---|---|---|---|---|---|
| 330 | 3.38 | 6.18 | 172 | 54.64 | 165 | 1265.04 |
| 500 | 7.84 | 21.41 | 253 | 205.12 | 246 | |
| 1000 | 33.05 | 237.10 | 500 | 2272.98 | 500 | |
| 1500 | 80.56 | 1010.76 | 750 | 9922.05 | 750 | |
| 2000 | 158.01 | 3422.38 | 1000 | | | |

TABLE 3.1: Elapsed time in seconds used for the generation of $\mathbf{M}_\alpha \in \mathcal{M}_{N\times N}(\mathbb{C})$, for different values of $N$ and different techniques. $\mathbf{M}_\alpha^{old}$ corresponds to the method described in Chapter 2; $\mathbf{M}_\alpha^{mp}$ uses the ADVANPIX function `mp`; $\mathbf{M}_\alpha^{vpa}$ uses `vpa`; and $\mathbf{M}_\alpha^{vpa}$ uses `hypergeom` without arbitrary precision. In the case of $\mathbf{M}_\alpha^{mp}$ and $\mathbf{M}_\alpha^{vpa}$, the number of digits to generate the matrices is also offered.

In our opinion, the results are quite revealing, allowing us to make some conclusions; indeed, for $N$ moderately large (e.g., $N = 500$), the method described in this chapter, together with ADVANPIX, is absolutely competitive; and reasonably fast, when combined with `vpa`. Furthermore, ADVANPIX, can be still recommended to generate larger matrices (e.g., $N = 1000$). On the other hand, even if, in principle, it is possible to generate very large matrices (e.g., $N = 1500$, $N = 2000$ or even larger sizes), it may be advisable to use in those case the method described in Chapter 2,

except when time is not an issue and the matrix needs to be generated only once or a few times. Finally, let us mention that the values of the discrepancy (3.36) between $\mathbf{M}_{\alpha}^{old}$ and $\mathbf{M}_{\alpha}^{np}$ for $N = 1000$, $N = 1500$ and $N = 2000$ are respectively $2.1238 \cdot 10^{-11}$, $5.1753 \cdot 10^{-11}$, and $8.2190 \cdot 10^{-11}$, i.e., of the order of $10^{-11}$.

# Chapter 4

# Other approaches for the fractional Laplacian on $\mathbb{R}$

In this chapter, we discuss other approaches to approximate numerically the fractional Laplacian on $\mathbb{R}$, using fast convolution. The methods thus developed, even if less accurate than those exposed in Chapters 2 and 3, are exceedingly fast. Furthermore, in some of them, it is possible to apply Richardson's extrapolation to improve the results.

## 4.1 Numerical convolution and the approximation of singular integrals

Let us consider two $N$-periodic sequences of complex numbers $u = \{u_r\}$ and $v = \{v_r\}$, with $r \in \mathbb{Z}$. Then, the convolution of $u$ and $v$, denoted by $u * v$, is a new sequence defined as

$$(u * v)_m \equiv \sum_{n=0}^{N-1} u_n v_{m-n}. \tag{4.1}$$

It is straightforward to check that $u * v$ is also $N$-periodic, and that $u * v = v * u$. A very important property of $u * v$ is that

$$(\widehat{u * v})_p = \widehat{u}_p \widehat{v}_p, \quad p = 0, \ldots, N-1, \tag{4.2}$$

This is a well-known theorem called the discrete convolution theorem, and it says in other words that the discrete Fourier transform of the convolution is just the product of the discrete Fourier transforms. Recall that, given an $N$-periodic sequence $u$, its discrete Fourier transform $\widehat{u}$ is also $N$-periodic, and is given by

$$\widehat{u}_p = \sum_{m=0}^{N-1} u_m e^{-\frac{2\pi i m p}{N}} \iff u_m = \frac{1}{N} \sum_{p=0}^{N-1} \widehat{u}_p e^{\frac{2\pi i m p}{N}}.$$

Then, the proof of (4.2) is straightforward (see for instance [9]):

$$\begin{aligned}(\widehat{u*v})_p &= \sum_{m=0}^{N-1}\left[\sum_{n=0}^{N-1} u_n v_{m-n}\right] e^{-\frac{2\pi i m p}{N}} = \sum_{n=0}^{N-1} u_n e^{-\frac{2\pi i n p}{N}} \sum_{m=0}^{N-1} v_{m-n} e^{-\frac{2\pi i (m-n) p}{N}} \\ &= \sum_{n=0}^{N-1} u_n e^{-\frac{2\pi i n p}{N}} \sum_{q=-n}^{N-n-1} v_q e^{-\frac{2\pi i q p}{N}} = \sum_{n=0}^{N-1} u_n e^{-\frac{2\pi i n p}{N}} \sum_{q=0}^{N-1} v_q e^{-\frac{2\pi i q p}{N}} \\ &= \widehat{u}_p \widehat{v}_p,\end{aligned}$$

where we have used that $v_q$ and $e^{-2\pi i q p/N}$ are $N$-periodic.

Observe that we need $\mathcal{O}(N^2)$ operations to compute directly (4.1). On the other hand, the computation of a discrete Fourier transform by means of the fast Fourier transform [3] algorithm requires $O(N \log N)$ operations. Since we need two fast Fourier transforms (FFT) and one inverse fast Fourier transform (IFFT) to compute (4.1) using (4.2), the total cost is also $O(N \log N)$ operations, which is much lower than $O(N^2)$. Therefore, this technique can be referred to as fast convolution.

In the case that the sequences $u$ and $v$ are not $N$-periodic, it is still possible tu compute (4.1) by means of fast convolution. Observe that, in (4.1), if $m = 0, \dots, N-1$, we need to know $\{u_0, \dots, u_{N-1}\}$ and $\{v_{-N+1}, \dots, v_{N-1}\}$. Therefore, we can regard them as $2N$-periodic by extending them respectively as follows [9]:

$$\widetilde{u}_r = \begin{cases} u_r, & r = 0, \dots, N-1, \\ 0, & r = N, \dots, 2N-1, \end{cases} \tag{4.3}$$

and

$$\widetilde{v}_r = \begin{cases} v_r, & r = 0, \dots, N-1, \\ 0, & n = N, \\ v_{r-2N}, & r = N+1, \dots, 2N-1, \end{cases} \tag{4.4}$$

and, for other values of $n$, imposing that $\widetilde{u}_{r+2N} = \widetilde{u}_r$ and $\widetilde{v}_{r+2N} = \widetilde{v}_r$. Then,

$$(u*v)_m \equiv \sum_{n=0}^{N-1} u_n v_{m-n} = \sum_{n=0}^{2N-1} \widetilde{u}_n \widetilde{v}_{m-n} = (\widetilde{u} * \widetilde{v})_m, \quad m = 0, \dots, N-1, \tag{4.5}$$

so, in order to obtain $u * v$, we extend $u$ and $v$ by means of (4.3) and (4.4), then compute $\widetilde{u} * \widetilde{v}$ via the fast convolution, and finally keep the first $N$ elements of $\widetilde{u} * \widetilde{v}$, and discard the last $N$. The global computation cost is again $\mathcal{O}(N \log N)$.

As we will see in this section, the application of the fast convolution enables to develop extremely fast methods to approximate the fractional Laplacian, although the price to pay is that they are less accurate than those offered in Chapters 2 and 3.

In order to introduce the approach that we will be using (and which takes ideas from [9]), let us start by presenting a simple example involving the computation of

a discrete convolution. More precisely, we consider the following integral:

$$G(s) = \int_0^\pi \frac{g(\eta)}{|s-\eta|^a} d\eta, \tag{4.6}$$

where $g(\eta) = f(\eta)/\eta^b$ and $a + b > -1$.

Let us consider first the simpler case $b = 0$, i.e., $g(\eta) \equiv f(\eta)$, which allows us to work with only one singularity in (4.6). In that case, we start by decomposing the domain into subintervals $I_n = [s_n, s_{n+1}]$, where $h = (s_N - s_0)/N = \pi/N$, $s_n = h\,n$, etc., and the function $g$ can be evaluated in analogously defined subintervals $I_m = [s_m, s_{m+1}]$ of the same length. For simplicity, $G$ is evaluated in the midpoint $s_{m+1/2}$ of every subinterval $I_m$, and so is $f$ in the midpoint of $I_n$, evaluation which we denote $f_{n+1/2}$. Observe that we are not using the definition of $s_n$ in (2.1); indeed, in Chapters 2 and 3, $s_n$ corresponds to what we call $s_{n+1/2}$ in this section; whereas we keep now $s_n$ to specify the ends of the subintervals. We think that this change of notation is justified, in order to clearly differentiate the ends of the intervals from their midpoints.

Bearing in mind the previous remarks, the following sum of integrals is constructed:

$$G\left(s_{m+\frac{1}{2}}\right) = \sum_{n=0}^{N-1} \int_{I_n} \frac{f(\eta)}{\left|s_{m+\frac{1}{2}} - \eta\right|^a} d\eta \approx \sum_{n=0}^{N-1} f_{n+\frac{1}{2}} \int_{I_n} \frac{d\eta}{\left|s_{m+\frac{1}{2}} - \eta\right|^a}.$$

Treating apart the previous integral with domain $I_n$, and applying the change of variable $\eta = s_n + (\lambda + 1/2)h$, with $s_n = hn$, we get:

$$M_{m-n} = \int_{s_n}^{s_{n+1}} \frac{d\eta}{\left|s_{m+\frac{1}{2}} - \eta\right|^a} = h^{1-a} \int_{-1/2}^{1/2} \frac{d\lambda}{|m-n-\lambda|^a}. \tag{4.7}$$

Therefore, we have the following discrete convolution

$$G\left(s_{m+\frac{1}{2}}\right) \approx \sum_{n=0}^{N-1} M_{m-n} f_{n+\frac{1}{2}} = (M * f)_m. \tag{4.8}$$

In order to apply the fast convolution to (4.8), we define, on the one hand, $\widetilde{f}_r$ according to (4.3):

$$\widetilde{f}_r = \begin{cases} f_{r+\frac{1}{2}}, & r = 0, \dots, N-1, \\ 0, & r = N, \dots, 2N-1. \end{cases} \tag{4.9}$$

On the other hand, setting $r = m - n$, we compute explicitly $M_r = M_{m-n}$:

$$\begin{aligned} M_r &= h^{1-a} \int_{-1/2}^{1/2} \frac{d\lambda}{|r-\lambda|^a} \\ &= \begin{cases} h^{1-a} \dfrac{2^a}{1-a}, & r = 0, \\ h^{1-a} \dfrac{\operatorname{sgn}(r+1/2)|r+1/2|^{1-a} - \operatorname{sgn}(r-1/2)|r-1/2|^{1-a}}{1-a}, & r \neq 0. \end{cases} \end{aligned} \tag{4.10}$$

Note that, since $r$ is an integer, when $r \neq 0$, we can simply write

$$M_r = h^{1-a} \frac{(|r|+1/2)^{1-a} - (|r|-1/2)^{1-a}}{1-a};$$

then, we define $\widetilde{M}_r$ according to (4.4):

$$\widetilde{M}_r = \begin{cases} M_r, & r = 0, \dots, N-1, \\ 0, & r = N, \\ M_{r-2N}, & r = N+1, \dots, 2N-1. \end{cases} \tag{4.11}$$

Finally, we apply (4.5), i.e., $(M * f)_m = (\widetilde{M} * \widetilde{f})_m$, for $m = 0, \dots, N-1$. From an implementational point of view, we compute $\widehat{f} = \mathrm{FFT}(\widetilde{f})$ and $\widehat{M} = \mathrm{FFT}(\widetilde{M})$, then multiply in the Fourier space, and calculate the IFFT of the result, keeping only the first $N$ elements of $\widetilde{M} * \widetilde{f}$.

| $j$ | $N$ | Error | $\log_2\left(\frac{E_{2^j}(\alpha)}{E_{2^{j+1}}(\alpha)}\right)$ |
|---|---|---|---|
| 4 | 16 | $7.183569 \cdot 10^{-3}$ | 1.98390 |
| 5 | 32 | $1.816047 \cdot 10^{-3}$ | 1.99421 |
| 6 | 64 | $4.558379 \cdot 10^{-4}$ | 1.98776 |
| 7 | 128 | $1.149302 \cdot 10^{-4}$ | 1.80143 |
| 8 | 256 | $3.297231 \cdot 10^{-5}$ | 2.20903 |
| 9 | 512 | $7.131227 \cdot 10^{-6}$ | |

TABLE 4.1: Errors obtained in discrete $L^\infty$-norm and convergence rate of (4.8) to (4.6), with $f(\eta) = e^{\sin(\eta)}$, $a = -0.5$ and $b = 0$.

We test the numerical convolution by taking $f(\eta) = \mathrm{e}^{\sin(\eta)}$, $\eta \in [0, \pi]$, and $a = -0.5$, $b = 0$. An adaptive Gauss-Legendre quadrature with $N = 2^8$ nodes (see e.g. [133], and [134], for an explicit routine) approximates the singular integrals corresponding to $G(s_m)$, and gives us the approximations that will be compared with those coming from the convolution scheme. In Table 4.1, we show the errors $E_{2^j}$ corresponding to $N = 2^j$, i.e., the differences between both approaches in discrete $L^\infty$-norm, and $\log_2(E_{2^j}(\alpha)/E_{2^{j+1}}(\alpha))$, for $j = 4, \dots, 8$, which gives us a convergence rate of order two.

We have also attempted the case $b \neq 0$, i.e., have considered

$$G\left(s_{m+\frac{1}{2}}\right) = \sum_{n=0}^{N-1} \int_{I_n} \frac{f(\eta)}{|\eta|^b \left|s_{m+\frac{1}{2}} - \eta\right|^a} d\eta. \tag{4.12}$$

In this case, in order to extract the term $f(\eta)/|\eta|^b$, we approximate it by its mean over each subinterval $I_n$; then, since it is still possible to evaluate $f$ at the midpoint of $I_n$, we further approximate the resulting integral by extracting $f_{n+1/2}$. More precisely, for $\eta \in I_n$,

$$\frac{f(\eta)}{|\eta|^b} \approx \frac{1}{h} \int_{I_n} \frac{f(\eta)}{|\eta|^b} d\eta \approx \frac{f_{n+\frac{1}{2}}}{h} \int_{I_n} \frac{1}{|\eta|^b} d\eta = f_{n+\frac{1}{2}} h^{-b} \int_0^1 \frac{d\lambda}{(n+\lambda)^b},$$

where we have applied the change of variable $\eta = s_n + \lambda h$. Therefore, we write

$$\begin{aligned} f_{n+\frac{1}{2}} h^{-b} \int_0^1 \frac{d\lambda}{(n+\lambda)^b} &= f_{n+\frac{1}{2}} \frac{h^{-b}((1+n)^{1-b} - n^{1-b})}{1-b} \\ &= f_{n+\frac{1}{2}} c_n, \end{aligned}$$

where

$$c_n = \frac{h^{-b}((1+n)^{1-b} - n^{1-b})}{1-b}.$$

Finally, bearing in mind (4.7) from the case $\beta = 0$, (4.12) is approximated as

$$\begin{aligned} G\left(s_{m+\frac{1}{2}}\right) &\approx \sum_{n=0}^{N-1} \int_{I_n} \left(\frac{1}{h} \int_{I_n} \frac{f(\eta)}{|\eta|^b} d\eta\right) \frac{d\eta}{\left|s_{m+\frac{1}{2}} - \eta\right|^a} \\ &\approx \sum_{n=0}^{N-1} c_n f_{n+\frac{1}{2}} \int_{I_n} \frac{d\eta}{\left|s_{m+\frac{1}{2}} - \eta\right|^a} \\ &= \sum_{n=0}^{N-1} c_n f_{n+\frac{1}{2}} M_{m-n} \\ &= (M * cf)_m. \end{aligned}$$

We have considered again $f(\eta) = e^{\sin(\eta)}$ as the testing function. Similarly as in Table 4.1, we show in Table 4.2 the errors $E_{2^j}$ and the convergence rate for $a = -0.75$ and $b = -0.15$.

## 4.2 Approximation of the fractional Laplacian using the fast convolution and midpoint rule

As said above, we want to develop a fast convolution formula applied to the fractional Laplacian. In order to do this, let us consider the integral definition given by (2.12). From this equation, it is possible to identify the corresponding kernel; in

| $j$ | $N$ | Error | $\log_2\left(\frac{E_{2j}(\alpha)}{E_{2j+1}(\alpha)}\right)$ |
|---|---|---|---|
| 4 | 16 | $8.723048 \cdot 10^{-3}$ | 2.00963 |
| 5 | 32 | $2.166259 \cdot 10^{-3}$ | 2.01458 |
| 6 | 64 | $5.361205 \cdot 10^{-4}$ | 2.01421 |
| 7 | 128 | $1.327166 \cdot 10^{-4}$ | 1.90892 |
| 8 | 256 | $3.534145 \cdot 10^{-5}$ | 2.11609 |
| 9 | 512 | $8.152231 \cdot 10^{-6}$ | |

TABLE 4.2: Errors obtained in discrete $L^\infty$-norm and convergence rate of (4.8) to (4.6) with $f(\eta) = e^{\sin(\eta)}$, $a = -0.75$ and $b = -0.15$.

what follows we will show how to deal with it. We take $\alpha \in (0,1) \cup (1,2)$, since the case $\alpha = 1$ is explicitly known: $(-\Delta)^{1/2}u = \mathcal{H}(u_x)$, i.e., we have the Hilbert transform of the derivative. Let us represent (2.12), $\alpha \neq 1$ as

$$(-\Delta)^{\alpha/2}u(s) = \frac{c_\alpha |\sin(s)|^{\alpha-1}}{L^\alpha \alpha(1-\alpha)} \int_0^\pi w(\eta) |\sin(s-\eta)|^{1-\alpha} \sin^\alpha(\eta) d\eta, \tag{4.13}$$

where $w(\eta) = \sin(\eta)u_{ss}(\eta) + 2\cos(\eta)u_s(\eta)$. To obtain a convolution sum, we decompose the integral over $[s_0, s_N] = [0, \pi]$ into a sum of integrals over subintervals $I_n = [s_n, s_{n+1}]$ of length $h = (s_N - s_0)/N = \pi/N$, as in the previous examples in Section 4.1. The evaluation of the function $u$, and consequently, the fractional Laplacian, will be carried out in the midpoint $s_{m+1/2}$ of each subinterval $I_m = [s_m, s_{m+1}]$:

$$(-\Delta)^{\alpha/2}u\left(s_{m+\frac{1}{2}}\right) = \frac{c_\alpha \left|\sin\left(s_{m+\frac{1}{2}}\right)\right|^{\alpha-1}}{L^\alpha \alpha(1-\alpha)} \sum_{n=0}^{N-1} \int_{s_n}^{s_{n+1}} w(\eta) \left|\sin\left(s_{m+\frac{1}{2}} - \eta\right)\right|^{1-\alpha} \sin^\alpha(\eta) d\eta. \tag{4.14}$$

The numerical approximation of the integral will be done as follows. First, we approximate the term $\sin^\alpha(\eta)$ by its mean on $I_n$, which is at its turn approximated by the midpoint rule:

$$\begin{aligned} \sin^\alpha(\eta) \approx \zeta_n &= \frac{1}{|I_n|} \int_{s_n}^{s_{n+1}} \sin^\alpha(\eta)\, d\eta \\ &\approx \frac{1}{h} \sin^\alpha\left(s_{n+\frac{1}{2}}\right)(s_{n+1} - s_n) \\ &= \sin^\alpha\left(h\left(n + \frac{1}{2}\right)\right); \end{aligned} \tag{4.15}$$

and, second, we approximate the integral of the kernel $|\sin(s_{m+1/2} - \eta)|^{1-\alpha}$,

$$K_{m-n} = \int_{s_n}^{s_{n+1}} \left|\sin\left(s_{m+\frac{1}{2}} - \eta\right)\right|^{1-\alpha} d\eta,$$

by distinguishing two cases, $r = m - n = 0$, and $r = m - n \neq 0$. If $r = m - n = 0$, we integrate the singularity directly, by performing the change of variable $\eta \equiv s_n + h(\lambda + 1/2)$, with $s_n = hn$, and using that $\sin(x) \sim x$, as $x \to 0$:

$$K_0 \approx h^{2-\alpha} \int_{-1/2}^{1/2} |m - n - \lambda|^{1-\alpha} d\lambda = h^{2-\alpha} \int_{-1/2}^{1/2} |-\lambda|^{1-\alpha} d\lambda = h^{2-\alpha} \frac{2^{\alpha-1}}{2-\alpha};$$

and if $r = m - n \neq 0$, we aproximate $K_r = K_{m-n}$ by the midpoint rule:

$$\begin{aligned} K_r &\approx \left|\sin\left(s_{m+\frac{1}{2}} - s_{n+\frac{1}{2}}\right)\right|^{1-\alpha} (s_{n+1} - s_n) \\ &= |\sin(h(m-n))|^{1-\alpha} h \\ &= |\sin(hr)|^{1-\alpha} h. \end{aligned}$$

Putting all together, (4.14) becomes

$$\begin{aligned} (-\Delta)^{\alpha/2} u\left(s_{m+\frac{1}{2}}\right) &\approx \frac{c_\alpha \left|\sin\left(s_{m+\frac{1}{2}}\right)\right|^{\alpha-1}}{L^\alpha \alpha(1-\alpha)} \sum_{n=0}^{N-1} K_{m-n} \zeta_n w_{n+\frac{1}{2}} \\ &= \frac{c_\alpha \left|\sin\left(s_{m+\frac{1}{2}}\right)\right|^{\alpha-1}}{L^\alpha \alpha(1-\alpha)} (K * \zeta w)_m, \end{aligned} \tag{4.16}$$

where $w_{n+1/2}$ denotes the evaluation of $w$ at the midpoint of the subinterval $I_n$. In order to apply the fast convolution to (4.16), we define, according to (4.3) and (4.4):

$$\widetilde{\zeta}_r \widetilde{w}_r = \begin{cases} \zeta_r w_{r+\frac{1}{2}}, & r = 0, \ldots, N-1, \\ 0, & r = N, \ldots, 2N-1, \end{cases}$$

$$\widetilde{K}_r = \begin{cases} K_r, & r = 0, \ldots, N-1, \\ 0, & r = N, \\ K_{r-2N}, & r = N+1, \ldots, 2N-1. \end{cases}$$

Then, we compute $\widetilde{K} * \widetilde{\zeta}\widetilde{w}$ with the fast convolution, and set $(K * \zeta w)_m = (\widetilde{K} * \widetilde{\zeta}\widetilde{w})_m$, for $m = 0, \ldots, N-1$.

### 4.2.1 Numerical experiments

In order to test this technique, we consider the functions $u_1(x)$ and $u_2(x)$ defined in Section 2.2:

$$u_1(x) = \frac{x^2 - 1}{x^2 + 1}, \quad \text{and} \quad u_2(x) = \frac{2x}{x^2 + 1}. \tag{4.17}$$

In this section, we use systematically $L = 1$ in the numerical experiments. Recall that $u_1(x) + iu_2(x)$, under the change of variable $x = \cot(s)$, with $L = 1$, becomes $e^{i2s}$, and its fractional Laplacian is given by (2.41). Thus, the exact and approximated solutions can be compared by computing the error in discrete $L^\infty$-norm, which we

denote $E_{2^j}$, for $N = 2^j$. On the left-hand of Figure 4.1 we plot the errors $E_{2^j}$ (from now on, using semi-logarithmic scale), and on the right-hand side, the convergence rate $\log_2\left(E_{2^j}(\alpha)/E_{2^{j+1}}(\alpha)\right)$ in terms of $\alpha$, for $\alpha = \{0.01,\ldots,0.99\} \cup \{1.01,\ldots,1.99\}$ (i.e., excluding the case $\alpha = 1$), taking $N = 2^j$ nodes, with $j = 7,8,\ldots,12$.

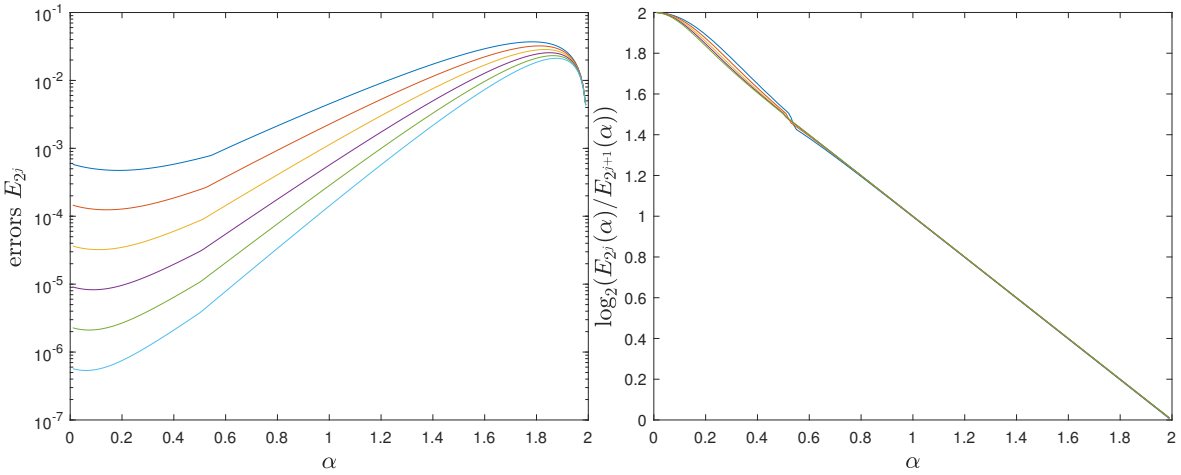


FIGURE 4.1: Left: Error for $(-\Delta)^{\alpha/2}e^{i2s}$. Right: Convergence rate. We have applied the fast convolution and midpoint rule, taking $\alpha \in \{0.01,\ldots,0.99\} \cup \{1.01,\ldots,1.99\}$, and $N = 2^j$ nodes, with $j = 7,8,\ldots,12$.

As a second example, we consider the Gaussian function

$$u_3(x) = \exp(-x^2) \tag{4.18}$$

from Section 2.2, whose fractional Laplacian is given by (2.43).

On the left-hand of Figure 4.2, we plot the errors and on the right-hand side, the convergence rate, taking $N = 2^j$ nodes, with $j = 7,8,\ldots,12$. Both Figure 4.1 and Figure 4.2 suggest that the convergence order is $2-\alpha$, i.e., we do not get the second order of Tables 4.1 and 4.2. In the following pages, we will show how to improve the convergence order.

## 4.3 Approximation of the fractional Laplacian using the fast convolution and a regular function

In order to improve the results of the previous section, we rewrite the integrand of (2.12), for $\alpha \neq 1$. More precisely, we construct a regular function of the form:

$$f(s,\eta) = \left(\sin(\eta)u_{ss}(\eta) + 2\cos(\eta)u_s(\eta)\right)\left|\frac{\sin(s-\eta)}{s-\eta}\right|^{1-\alpha}\left(\frac{\sin(\eta)}{\eta}\right)^{\alpha}. \tag{4.19}$$

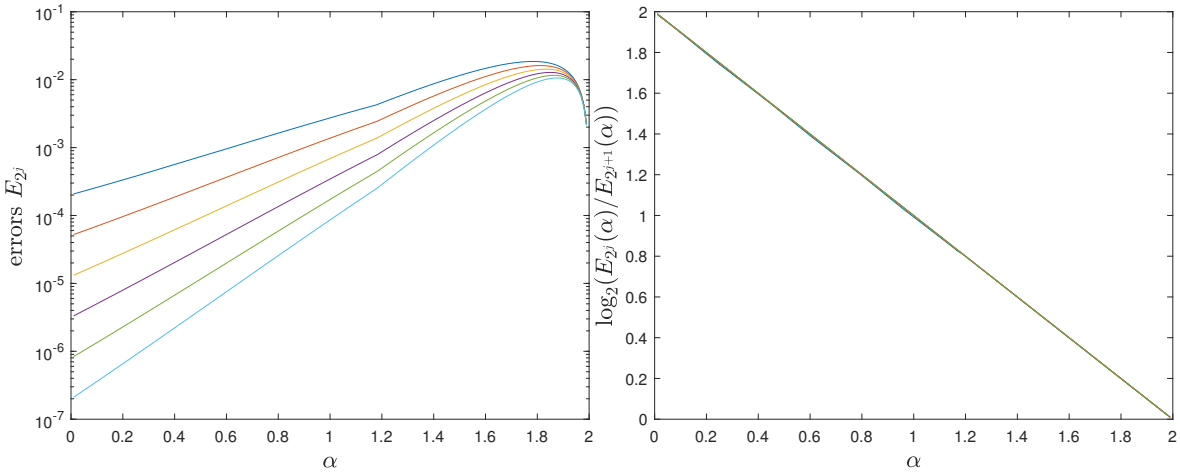


FIGURE 4.2: Left: Errors for $(-\Delta)^{\alpha/2}e^{-x^2}$. Right: Convergence rate. We have applied the fast convolution and midpoint rule, taking $\alpha \in \{0.01,\ldots,0.99\} \cup \{1.01,\ldots,1.99\}$, and $N = 2^j$ nodes, with $j = 7,8,\ldots,12$.

Hence, when $\alpha \neq 1$, (2.12) evaluated at the midpoints $s_{m+1/2}$ of $I_m$ is expressed as

$$(-\Delta)^{\alpha/2}u\left(s_{m+\frac{1}{2}}\right) = \frac{c_\alpha \left|\sin\left(s_{m+\frac{1}{2}}\right)\right|^{\alpha-1}}{L^\alpha \alpha(1-\alpha)} \sum_{n=0}^{N-1} \int_{s_n}^{s_{n+1}} f\left(s_{m+\frac{1}{2}},\eta\right) \left|s_{m+\frac{1}{2}}-\eta\right|^{1-\alpha} \eta^\alpha d\eta.$$

We proceed in a similar way as we did for (4.14). We first approximate $f\left(s_{m+1/2},\eta\right)$ in $I_n = [s_n, s_{n+1}]$ by its value at the midpoint of $I_n$, i.e.,

$$f\left(s_{m+\frac{1}{2}}, s_{n+\frac{1}{2}}\right) \approx f\left(s_{m+\frac{1}{2}},\eta\right);$$

hence,

$$(-\Delta)^{\alpha/2}u\left(s_{m+\frac{1}{2}}\right) \approx \frac{c_\alpha \left|\sin\left(s_{m+\frac{1}{2}}\right)\right|^{\alpha-1}}{L^\alpha \alpha(1-\alpha)} f\left(s_{m+\frac{1}{2}}, s_{n+\frac{1}{2}}\right) \sum_{n=0}^{N-1} \int_{s_n}^{s_{n+1}} \left|s_{m+\frac{1}{2}}-\eta\right|^{1-\alpha} \eta^\alpha d\eta. \quad (4.20)$$

To further proceed, we approximate the term $\eta^\alpha$ by its mean on $I_n$:

$$\eta^\alpha \approx \zeta_n = \frac{1}{h}\int_{s_n}^{s_{n+1}} \eta^\alpha d\eta = \frac{s_{n+1}^{1+\alpha} - s_n^{1+\alpha}}{(1+\alpha)h} = h^\alpha \left[\frac{(n+1)^{1+\alpha} - n^{1+\alpha}}{1+\alpha}\right], \quad \eta \in I_n;$$

and compute the integral of the kernel $|s_{m+1/2} - \eta|^{1-\alpha}$,

$$K_{m-n} = \int_{s_n}^{s_{n+1}} \left| s_{m+\frac{1}{2}} - \eta \right|^{1-\alpha} d\eta = h^{2-\alpha} \int_{-1/2}^{1/2} |m - n - \lambda|^{1-\alpha} d\lambda;$$

writing $r = m - n$, we obtain an expression identical to (4.10), after replacing $a$ by $\alpha - 1$:

$$K_r = \begin{cases} h^{2-\alpha} \dfrac{2^{\alpha-1}}{2-\alpha}, & r = 0, \\ h^{2-\alpha} \dfrac{\operatorname{sgn}(r+1/2)|r+1/2|^{2-\alpha} - \operatorname{sgn}(r-1/2)|r-1/2|^{2-\alpha}}{2-\alpha}, & r \neq 0. \end{cases}$$

Therefore, since $r$ is an integer, when $r \neq 0$, we can write

$$K_r = h^{2-\alpha} \frac{(|r| + 1/2)^{2-\alpha} - (|r| - 1/2)^{2-\alpha}}{2-\alpha}.$$

Putting all together, (4.20) becomes

$$(-\Delta)^{\alpha/2} u\left(s_{m+\frac{1}{2}}\right) \approx \frac{c_\alpha \left|\sin\left(s_{m+\frac{1}{2}}\right)\right|^{\alpha-1}}{L^\alpha \alpha(1-\alpha)} \sum_{n=0}^{N-1} f\left(s_{m+\frac{1}{2}}, s_{n+\frac{1}{2}}\right) K_{m-n} \zeta_n. \tag{4.21}$$

However, in this form, we cannot apply the fast convolution, so we have to expand $f(s_{m+1/2}, s_{n+1/2})$ according to its definition in (4.19) and gather together the terms bearing $n$ and those bearing $m - n$ to get the following expression:

$$\begin{aligned}(-\Delta)^{\alpha/2} u\left(s_{m+\frac{1}{2}}\right) \approx \frac{c_\alpha \left|\sin\left(s_{m+\frac{1}{2}}\right)\right|^{\alpha-1}}{L^\alpha \alpha(1-\alpha)} \sum_{n=0}^{N-1} \left|\frac{\sin(h(m-n))}{h(m-n)}\right|^{1-\alpha} K_{m-n} \\ (\sin(s_{n+\frac{1}{2}}) u_{ss}(s_{n+\frac{1}{2}}) + 2\cos(s_{n+\frac{1}{2}}) u_s(s_{n+\frac{1}{2}})) \left(\frac{\sin(s_{n+\frac{1}{2}})}{s_{n+\frac{1}{2}}}\right)^\alpha \zeta_n,\end{aligned}$$

where we have used that $s_{m+1/2} - s_{n+1/2} = h(m-n)$. Finally, defining,

$$\begin{aligned} g_r &\equiv \left(\sin\left(s_{r+\frac{1}{2}}\right) u_{ss}\left(s_{r+\frac{1}{2}}\right) + 2\cos\left(s_{r+\frac{1}{2}}\right) u_s\left(s_{r+\frac{1}{2}}\right)\right) \left(\frac{\sin\left(s_{r+\frac{1}{2}}\right)}{s_{r+\frac{1}{2}}}\right)^\alpha \zeta_r, \\ L_r &\equiv \begin{cases} K_r, & r = 0, \\ \left|\dfrac{\sin(hr)}{hr}\right|^{1-\alpha} K_r, & r \neq 0, \end{cases} \end{aligned} \tag{4.22}$$

where $s_{r+1/2} = h(r+1/2)$, and constructing $\widetilde{L}_r$ and $\widetilde{g}_r$ as in (4.3) and (4.4), we conclude that

$$(-\Delta)^{\alpha/2} u\left(s_{m+\frac{1}{2}}\right) \approx \frac{c_\alpha \left|\sin\left(s_{m+\frac{1}{2}}\right)\right|^{\alpha-1}}{L^\alpha \alpha(1-\alpha)} (\widetilde{L} \star \widetilde{g})_m. \tag{4.23}$$

### 4.3.1 Numerical experiments

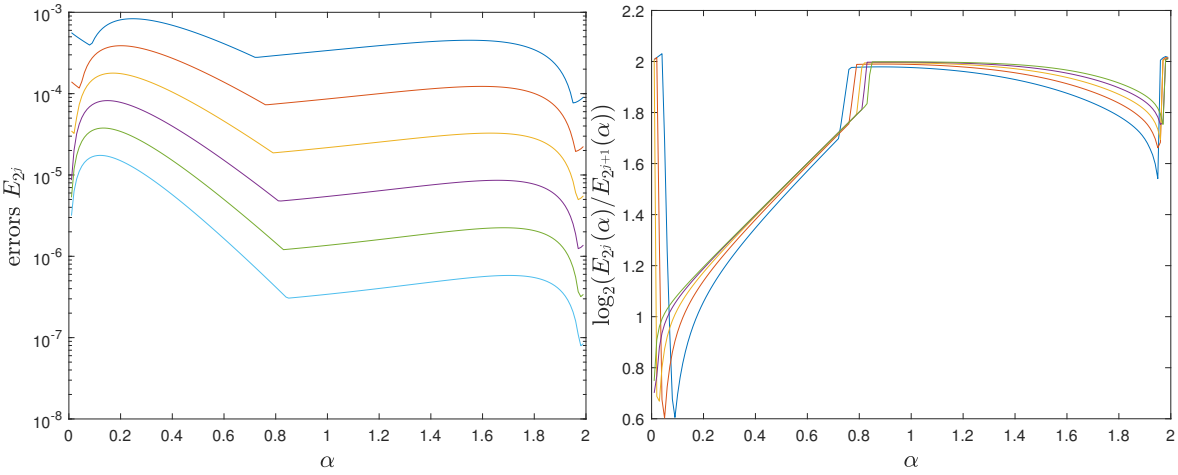


FIGURE 4.3: Left: Errors for $(-\Delta)^{\alpha/2} e^{i2s}$. Right: Convergence rate. We have applied the fast convolution scheme, using a regular function as the integrand, taking $\alpha \in \{0.01, \ldots, 0.99\} \cup \{1.01, \ldots, 1.99\}$, and $N = 2^j$ nodes, with $j = 7, 8, \ldots, 12$.

In order to test (4.23) numerically, we consider again the functions $u_1(x) + iu_2(x) = e^{i2s}$ and $u_3(x) = \exp(-x^2)$ defined in (4.17) and (4.18), and also used in Section 2.2. We have computed the errors in the discrete $L^\infty$-norm, for $N = 2^j$, with $j = 7, 8, \ldots, 12$, and the convergence rate. On the left-hand side of Figure 4.3, the errors $E_{2^j}$ are plotted against $\alpha$. On the right-hand side of Figure 2.1, the convergence rate is plotted against $\alpha$, resembling to be $1 + \alpha$ when $\alpha \in (0, 1)$ and 2, when $\alpha \in (1, 2)$.

In the case of $u_3(x)$, on the left-hand side of Figure 4.4, the errors $E_{2^j}$ are plotted against $\alpha$. On the right-hand side of Figure 4.4, the convergence rate is plotted against $\alpha$. In this case, $N = 2^j$, with $j = 7, 18, \ldots, 12$. Figures 4.3 and 4.4 suggest that the convergence rate for this technique is approximately $1 + \alpha$ when $\alpha \in (0, 1)$ and 2, when $\alpha \in (1, 2)$. Although these results are better than those in Figures 4.1 and 4.2, they can still be improved, as we will see in the following section.

## 4.4 Approximation of the fractional Laplacian using the fast convolution and extrapolation

Let us consider the partition of $[0, \pi]$ given by the nodes $s_n^{(j)}$ in (2.3) (see also [100]):

$$s_n^{(j)} = \frac{\pi(2n+1)}{2^{j+1}N}, \quad 0 \le n \le 2^j N - 1; \tag{4.24}$$

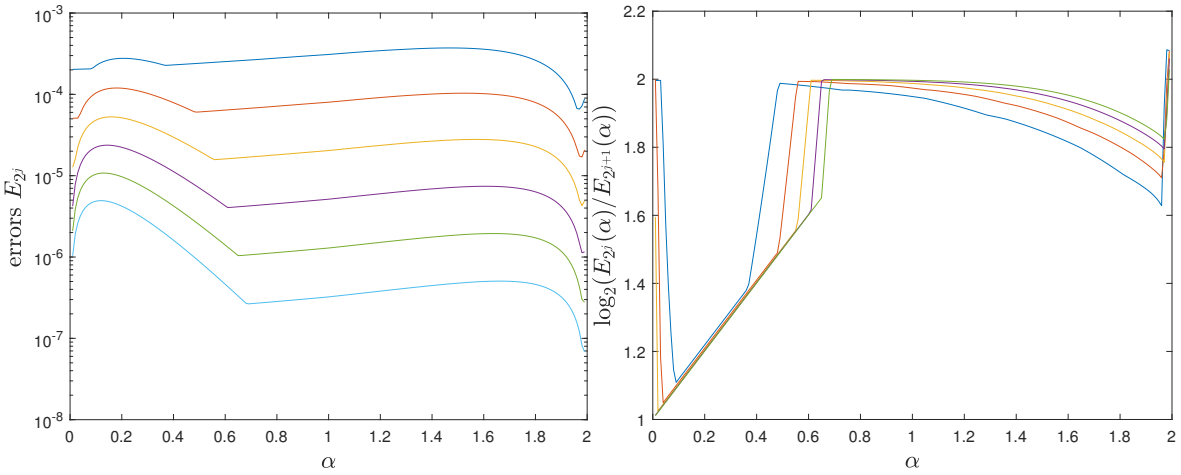


FIGURE 4.4: Left: Errors for $(-\Delta)^{\alpha/2}e^{-x^2}$. Right: Convergence rate. We have applied the fast convolution scheme, using a regular function as the integrand, taking $\alpha \in \{0.01, \dots, 0.99\} \cup \{1.01, \dots, 1.99\}$, and $N = 2^j$ nodes, with $j = 7, 8, \dots, 12$.

observe that there are no two nodes $s_n^{(j)}$ identical. Moreover, when $j = 0$, the nodes ${s_n}^{(0)}$ are precisely the middle points of the intervals $I_n = [s_n, s_{n+1}]$, $h = \pi/N$, used in the previous sections, i.e., ${s_n}^{(0)} = s_{n+1/2}$. On the other hand, when $j \in \mathbb{N}$,

$$\begin{aligned} s_{2n}^{(1)} &= s_{n+1/4},\ s_{2n+1}^{(1)} = s_{n+3/4}, \\ s_{4n}^{(2)} &= s_{n+1/8},\ s_{4n+1}^{(2)} = s_{n+3/8},\ s_{4n+2}^{(2)} = s_{n+5/8},\ s_{4n+3}^{(2)} = s_{n+7/8}, \end{aligned}$$

etc., i.e., each $I_n$ contains one node $s_n^{(0)}$, two nodes $s_n^{(1)}$, four nodes $s_n^{(2)}$, etc. Therefore, the nodes ${s_n}^{(j+1)}$ can be regarded as a refinement of ${s_n}^{(j)}$, and in general of $s_{n+1/2}$.

Let us construct the following function, which resembles (4.19), except that there is no appearance of $\eta^\alpha$:

$$f(s,\eta) = (\sin(\eta)u_{ss}(\eta) + 2\cos(\eta)u_s(\eta)) \left|\frac{\sin(s-\eta)}{s-\eta}\right|^{1-\alpha} \sin^\alpha(\eta). \tag{4.25}$$

Thus, we obtain the following expression for the fractional Laplacian, when $\alpha \neq 1$:

$$(-\Delta)^{\alpha/2}u(s) = \frac{c_\alpha |\sin(s)|^{\alpha-1}}{L^\alpha \alpha(1-\alpha)} \int_0^\pi f(s,\eta)|s-\eta|^{1-\alpha} d\eta. \tag{4.26}$$

In what follows, we will obtain an approximation of (4.26) that can be computed by means of the fast convolution. More precisely, the fact that each node of the form ${s_{2n}}^{(1)}$ is the middle point of $[s_n, s_{n+1/2}]$, and each node of the form ${s_{2n+1}}^{(1)}$ is the

middle point of $[s_{n+1/2}, s_{n+1}]$, suggests expressing (4.26) as

$$(-\Delta)^{\alpha/2} u\left(s_{m+\frac{1}{2}}\right) = \frac{c_\alpha \left|\sin\left(s_{m+\frac{1}{2}}\right)\right|^{\alpha-1}}{L^\alpha \alpha(1-\alpha)} \sum_{n=0}^{N-1} \left[ \int_{s_n}^{s_{n+\frac{1}{2}}} f\left(s_{m+\frac{1}{2}}, \eta\right) \left|s_{m+\frac{1}{2}} - \eta\right|^{1-\alpha} d\eta + \int_{s_{n+\frac{1}{2}}}^{s_{n+1}} f\left(s_{m+\frac{1}{2}}, \eta\right) \left|s_{m+\frac{1}{2}} - \eta\right|^{1-\alpha} d\eta \right], \tag{4.27}$$

i.e., we have split $[0, \pi]$ in intervals $I_n$; and each interval $I_n$, in two subintervals $[s_n, s_{n+1/2}]$ and $[s_{n+1/2}, s_{n+1}]$. Then, we approximate $f(s_{m+1/2}, \eta)$ in those intervals precisely by its values at the middle points:

$$f\left(s_{m+1/2}, s_{2n}^{(1)}\right) \approx f(s_{m+1/2}, \eta), \quad \eta \in [s_n, s_{n+1/2}],$$
$$f\left(s_{m+1/2}, s_{2n+1}^{(1)}\right) \approx f(s_{m+1/2}, \eta), \quad \eta \in [s_{n+1/2}, s_{n+1}],$$

to get

$$(-\Delta)^{\alpha/2} u\left(s_{m+\frac{1}{2}}\right) \approx \frac{c_\alpha \left|\sin\left(s_{m+\frac{1}{2}}\right)\right|^{\alpha-1}}{L^\alpha \alpha(1-\alpha)} \left[ \sum_{n=0}^{N-1} f\left(s_{m+\frac{1}{2}}, s_{2n}^{(1)}\right) \int_{s_n}^{s_{n+\frac{1}{2}}} \left|s_{m+\frac{1}{2}} - \eta\right|^{1-\alpha} d\eta + \sum_{n=0}^{N-1} f\left(s_{m+\frac{1}{2}}, s_{2n+1}^{(1)}\right) \int_{s_{n+\frac{1}{2}}}^{s_{n+1}} \left|s_{m+\frac{1}{2}} - \eta\right|^{1-\alpha} d\eta \right]. \tag{4.28}$$

The integrals in (4.28) can be computed exactly:

$$\begin{aligned} K_{0,m-n}^{(1)} &= \int_{s_n}^{s_{n+\frac{1}{2}}} \left|s_{m+\frac{1}{2}} - \eta\right|^{1-\alpha} d\eta \\ &= h^{2-\alpha} \frac{\operatorname{sgn}(m-n+1/2)|m-n+1/2|^{2-\alpha} - \operatorname{sgn}(m-n)|m-n|^{2-\alpha}}{2-\alpha}, \\ K_{1,m-n}^{(1)} &= \int_{s_{n+\frac{1}{2}}}^{s_{n+1}} \left|s_{m+\frac{1}{2}} - \eta\right|^{1-\alpha} d\eta \\ &= h^{2-\alpha} \frac{\operatorname{sgn}(m-n)|m-n|^{2-\alpha} - \operatorname{sgn}(m-n-1/2)|m-n-1/2|^{2-\alpha}}{2-\alpha}; \end{aligned}$$

observe that $K^{(1)}_{0,n-m} = K^{(1)}_{1,m-n}$. Then, (4.28) becomes

$$(-\Delta)^{\alpha/2} u\left(s_{m+\frac{1}{2}}\right) \approx \frac{c_\alpha \left|\sin\left(s_{m+\frac{1}{2}}\right)\right|^{\alpha-1}}{L^\alpha \alpha(1-\alpha)} \left[\sum_{n=0}^{N-1} f\left(s_{m+\frac{1}{2}}, s^{(1)}_{2n}\right) K^{(1)}_{0,m-n} + \sum_{n=0}^{N-1} f\left(s_{m+\frac{1}{2}}, s^{(1)}_{2n+1}\right) K^{(1)}_{1,m-n}\right]. \tag{4.29}$$

In order to be able to apply the fast convolution in the sums, we operate as in (4.21), i.e., expand $f(s_{m+1/2}, {s_{2n}}^{(1)})$ and $f(s_{m+1/2}, {s_{2n+1}}^{(1)})$ according to their definition in (4.25) and gather together the terms bearing $n$ and those bearing $m-n$, which, similarly as in (4.22), leads us to define

$$\begin{aligned}
g^{(1)}_{0,r} &\equiv \sin^{1+\alpha}\left(s^{(1)}_{2r}\right) u_{ss}\left(s^{(1)}_{2r}\right) + 2\sin^\alpha\left(s^{(1)}_{2r}\right)\cos\left(s^{(1)}_{2r}\right) u_s\left(s^{(1)}_{2r}\right), \qquad (4.30)\\
L^{(1)}_{0,r} &\equiv \left|\frac{\sin(h(r+1/4))}{h(r+1/4)}\right|^{1-\alpha} K^{(1)}_{1,r}\\
&= h\left|\frac{\sin(h(r+1/4))}{r+1/4}\right|^{1-\alpha} \frac{\mathrm{sgn}(r+1/2)|r+1/2|^{2-\alpha} - \mathrm{sgn}(r)|r|^{2-\alpha}}{2-\alpha},\\
g^{(1)}_{1,r} &\equiv \sin^{1+\alpha}\left(s^{(1)}_{2r+1}\right) u_{ss}\left(s^{(1)}_{2r+1}\right) + 2\sin^\alpha\left(s^{(1)}_{2r+1}\right)\cos\left(s^{(1)}_{2r+1}\right) u_s\left(s^{(1)}_{2r+1}\right), \qquad (4.31)\\
L^{(1)}_{1,r} &\equiv \left|\frac{\sin(h(r-1/4))}{h(r-1/4)}\right|^{1-\alpha} K^{(1)}_{2,r}\\
&= h\left|\frac{\sin(h(r-1/4))}{r-1/4}\right|^{1-\alpha} \frac{\mathrm{sgn}(r)|r|^{2-\alpha} - \mathrm{sgn}(r-1/2)|r-1/2|^{2-\alpha}}{2-\alpha},
\end{aligned}$$

where, ${s_{2r}}^{(1)} = s_{r+1/4} = h(r+1/4)$, ${s_{2r+1}}^{(1)} = s_{r+3/4} = h(r+3/4)$, and in the definitions of ${K_{0,m-n}}^{(1)}$ and ${K_{1,m-n}}^{(1)}$, we have replaced $m-n$ by $r$. In order to generate efficiently ${g_{0,r}}^{(1)}$ and ${g_{1,r}}^{(1)}$, for $0 \le r \le N-1$, we evaluate

$$g(s) = \sin^{1+\alpha}(s) u_{ss}(s) + 2\sin^\alpha(s)\cos(s) u_s(s) \tag{4.32}$$

in all the nodes ${s_r}^{(1)}$, for $0 \le r \le 2N-1$; the values in odd and even positions form ${g_0}^{(1)}$, and ${g_1}^{(1)}$, respectively. Note also that $L_{1,r} = L_{0,-r}$. Therefore, (4.29) can be expressed as the sum of two actual convolutions:

$$(-\Delta)^{\alpha/2} u\left(s_{m+\frac{1}{2}}\right) \approx \frac{c_\alpha \left|\sin\left(s_{m+\frac{1}{2}}\right)\right|^{\alpha-1}}{L^\alpha \alpha(1-\alpha)} \left[\sum_{n=0}^{N-1} L^{(1)}_{0,m-n} g^{(1)}_{0,n} + \sum_{n=0}^{N-1} L^{(1)}_{1,m-n} g^{(1)}_{1,n}\right]. \tag{4.33}$$

Finally, constructing ${\widetilde{g}_0}^{(1)}$, ${\widetilde{L}_0}^{(1)}$, ${\widetilde{g}_1}^{(1)}$ and ${\widetilde{L}_1}^{(1)}$, as in (4.3) and (4.4):

$$(-\Delta)^{\alpha/2} u\left(s_{m+\frac{1}{2}}\right) \approx \frac{c_\alpha \left|\sin\left(s_{m+\frac{1}{2}}\right)\right|^{\alpha-1}}{L^\alpha \alpha(1-\alpha)} \left[\left(\widetilde{L}^{(1)}_0 \star \widetilde{g}^{(1)}_0\right)_m + \left(\widetilde{L}^{(1)}_1 \star \widetilde{g}^{(1)}_1\right)_m\right].$$

Observe that, in principle, we need to compute the FFT of $\widetilde{L}_0{}^{(1)}$, $\widetilde{g}_0{}^{(1)}$, $\widetilde{L}_1{}^{(1)}$ and $\widetilde{g}_1{}^{(1)}$, but only one single IFFT:

$$\widetilde{L}_0^{(1)} \star \widetilde{g}_0^{(1)} + \widetilde{L}_1^{(1)} \star \widetilde{g}_1^{(1)} = \mathrm{IFFT}(\mathrm{FFT}(\widetilde{L}_0^{(1)})\,\mathrm{FFT}(\widetilde{g}_0^{(1)}) + \mathrm{FFT}(\widetilde{L}_1^{(1)})\,\mathrm{FFT}(\widetilde{g}_1^{(1)})).$$

Moreover, in some programming languages like MATLAB it is possible to compute the four FFTs in parallel, after, for instance, creating a matrix whose respective columns are precisely $\widetilde{L}_0{}^{(1)}$, $\widetilde{g}_0{}^{(1)}$, $\widetilde{L}_1{}^{(1)}$ and $\widetilde{g}_1{}^{(1)}$. Furthermore,

$$\mathrm{FFT}(\widetilde{L}_1^{(1)}) = \overline{\mathrm{FFT}(\widetilde{L}_0^{(1)})};$$

so, the actual number of FFTs is reduced to three, and, depending on the structure of $u$, it might be possible to obtain $\mathrm{FFT}(\widetilde{g}_1{}^{(1)})$ from $\mathrm{FFT}(\widetilde{g}_0{}^{(1)})$ and vice versa.

### 4.4.1 Refining the mesh

The advantage of the procedure just explained is that we can refine the mesh, while approximating the fractional Laplacian in the very same nodes $s_{m+1/2}$. For instance, if we divide $[0,\pi]$ in four equally-lengthed subintervals,

$$\begin{aligned}(-\Delta)^{\alpha/2}u\left(s_{m+\frac{1}{2}}\right) = \frac{c_\alpha\left|\sin\left(s_{m+\frac{1}{2}}\right)\right|^{\alpha-1}}{L^\alpha\alpha(1-\alpha)}\sum_{n=0}^{N-1}\Bigg[&\int_{s_n}^{s_{n+\frac{1}{4}}} f\left(s_{m+\frac{1}{2}},\eta\right)\left|s_{m+\frac{1}{2}}-\eta\right|^{1-\alpha}d\eta\\ &+\int_{s_{n+\frac{1}{4}}}^{s_{n+\frac{1}{2}}} f\left(s_{m+\frac{1}{2}},\eta\right)\left|s_{m+\frac{1}{2}}-\eta\right|^{1-\alpha}d\eta\\ &+\int_{s_{n+\frac{1}{2}}}^{s_{n+\frac{3}{4}}} f\left(s_{m+\frac{1}{2}},\eta\right)\left|s_{m+\frac{1}{2}}-\eta\right|^{1-\alpha}d\eta\\ &+\int_{s_{n+\frac{3}{4}}}^{s_{n+1}} f\left(s_{m+\frac{1}{2}},\eta\right)\left|s_{m+\frac{1}{2}}-\eta\right|^{1-\alpha}d\eta\Bigg].\end{aligned}$$

Therefore, the equivalent of (4.28) is

$$\begin{aligned}(-\Delta)^{\alpha/2}u\left(s_{m+\frac{1}{2}}\right) \approx \frac{c_\alpha\left|\sin\left(s_{m+\frac{1}{2}}\right)\right|^{\alpha-1}}{L^\alpha\alpha(1-\alpha)}\Bigg[&\sum_{n=0}^{N-1} f\left(s_{m+\frac{1}{2}},s_{4n}^{(2)}\right)\int_{s_n}^{s_{n+\frac{1}{4}}}\left|s_{m+\frac{1}{2}}-\eta\right|^{1-\alpha}d\eta\\ &+\sum_{n=0}^{N-1} f\left(s_{m+\frac{1}{2}},s_{4n+1}^{(2)}\right)\int_{s_{n+\frac{1}{4}}}^{s_{n+\frac{1}{2}}}\left|s_{m+\frac{1}{2}}-\eta\right|^{1-\alpha}d\eta\\ &+\sum_{n=0}^{N-1} f\left(s_{m+\frac{1}{2}},s_{4n+2}^{(2)}\right)\int_{s_{n+\frac{1}{2}}}^{s_{n+\frac{3}{4}}}\left|s_{m+\frac{1}{2}}-\eta\right|^{1-\alpha}d\eta\\ &+\sum_{n=0}^{N-1} f\left(s_{m+\frac{1}{2}},s_{4n+3}^{(2)}\right)\int_{s_{n+\frac{3}{4}}}^{s_{n+1}}\left|s_{m+\frac{1}{2}}-\eta\right|^{1-\alpha}d\eta\Bigg],\end{aligned}$$

where ${s_{4n}}^{(2)}$, ${s_{4n+1}}^{(2)}$, ${s_{4n+2}}^{(2)}$ and ${s_{4n+3}}^{(2)}$ are respectively the middle points of $[s_n, s_{n+1/4}]$, $[s_{n+1/4}, s_{n+1/2}]$, $[s_{n+1/2}, s_{n+3/4}]$, and $[s_{n+3/4}, s_{n+1}]$. Using a more compact notation:

$$(-\Delta)^{\alpha/2}u\left(s_{m+\frac{1}{2}}\right) \approx \frac{c_\alpha \left|\sin\left(s_{m+\frac{1}{2}}\right)\right|^{\alpha-1}}{L^\alpha \alpha(1-\alpha)} \sum_{n=0}^{N-1}\sum_{p=0}^{3} f\left(s_{m+\frac{1}{2}}, s^{(2)}_{4n+p}\right) \int_{s_{n+\frac{p}{4}}}^{s_{n+\frac{p+1}{4}}} \left|s_{p+\frac{1}{2}} - \eta\right|^{1-\alpha} d\eta, \quad (4.34)$$

where the integrals can be explicitly computed:

$$\begin{aligned} K^{(2)}_{p,m-n} &= \int_{s_{n+\frac{p}{4}}}^{s_{n+\frac{p+1}{4}}} \left|s_{m+\frac{1}{2}} - \eta\right|^{1-\alpha} d\eta \\ &= h^{2-\alpha}\frac{\operatorname{sgn}(m-n+1/2-p/4)|m-n+1/2-p/4|^{2-\alpha}}{2-\alpha} \\ &\quad - h^{2-\alpha}\frac{\operatorname{sgn}(m-n+1/2-(p+1)/4)|m-n+1/2-(p+1)/4|^{2-\alpha}}{2-\alpha}. \end{aligned}$$

Introducing them into (4.34) and expanding $f(s_{m+1/2}, {s_{4n+p}}^{(2)})$ according to its definition in (4.19), (4.34) is reduced to the sum of four convolutions:

$$\begin{aligned} (-\Delta)^{\alpha/2}u\left(s_{m+\frac{1}{2}}\right) &\approx \frac{c_\alpha \left|\sin\left(s_{m+\frac{1}{2}}\right)\right|^{\alpha-1}}{L^\alpha \alpha(1-\alpha)} \sum_{n=0}^{N-1}\sum_{p=0}^{3} f\left(s_{m+\frac{1}{2}}, s^{(2)}_{4n+p}\right) K^{(2)}_{p,m-n} \\ &= \frac{c_\alpha \left|\sin\left(s_{m+\frac{1}{2}}\right)\right|^{\alpha-1}}{L^\alpha \alpha(1-\alpha)} \sum_{p=0}^{3}\left[\sum_{n=0}^{N-1} L^{(2)}_{p,m-n} g^{(2)}_{p,n}\right], \end{aligned} \quad (4.35)$$

where

$$\begin{aligned} g^{(2)}_{p,r} &\equiv \sin^{1+\alpha}\left(s^{(2)}_{4r+p}\right) u_{ss}\left(s^{(2)}_{4r+p}\right) + 2\sin^{\alpha}\left(s^{(2)}_{4r+p}\right)\cos\left(s^{(2)}_{4r+p}\right) u_s\left(s^{(2)}_{4r+p}\right), \\ L^{(2)}_{p,r} &\equiv \left|\frac{\sin(h(r+1/2-(2p+1)/8))}{h(r+1/2-(2p+1)/8)}\right|^{1-\alpha} K^{(2)}_{p,r} \\ &= h\left|\frac{\sin(h(r+1/2-(2p+1)/8))}{r+1/2-(2p+1)/8}\right|^{1-\alpha}\left[\frac{\operatorname{sgn}(r+1/2-p/4)|r+1/2-p/4|^{2-\alpha}}{2-\alpha}\right. \\ &\quad \left. - \frac{\operatorname{sgn}(r+1/2-(p+1)/4)|r+1/2-(p+1)/4|^{2-\alpha}}{2-\alpha}\right]. \end{aligned}$$

with ${s_{4r+p}}^{(2)} = s_{r+(2p+1)/8}$. In order to generate efficiently ${g_p}^{(2)}$, one possibility is to evaluate $g(s)$ in (4.32) in all the nodes ${s_r}^{(2)}$, for $0 \le r \le 4N-1$; the first, fifth... values form ${g_0}^{(2)}$, the second, sixth... values form ${g_1}^{(2)}$, etc. On the other hand, it is immediate to check that ${L_{3,r}}^{(2)} = {L_{0,-r}}^{(2)}$ and ${L_{2,r}}^{(2)} = {L_{1,-r}}^{(2)}$. Then, we construct

$\widetilde{g}_0{}^{(2)}$, $\widetilde{L}_0{}^{(2)}$, $\widetilde{g}_1{}^{(2)}$, $\widetilde{L}_1{}^{(2)}$, $\widetilde{g}_2{}^{(2)}$, $\widetilde{L}_2{}^{(2)}$, $\widetilde{g}_3{}^{(2)}$ and $\widetilde{L}_3{}^{(2)}$, as in (4.3) and (4.4), to get

$$(-\Delta)^{\alpha/2}u\left(s_{m+\frac{1}{2}}\right) \approx \frac{c_\alpha \left|\sin\left(s_{m+\frac{1}{2}}\right)\right|^{\alpha-1}}{L^\alpha \alpha(1-\alpha)}\Big[\left(\widetilde{L}_0^{(2)} \star \widetilde{g}_0^{(2)}\right)_m + \left(\widetilde{L}_1^{(2)} \star \widetilde{g}_1^{(2)}\right)_m + \left(\widetilde{L}_2^{(2)} \star \widetilde{g}_2^{(2)}\right)_m + \left(\widetilde{L}_3^{(2)} \star \widetilde{g}_3^{(2)}\right)_m\Big].$$

Moreover, bearing in mind that

$$\mathrm{FFT}(\widetilde{L}_3^{(2)}) = \overline{\mathrm{FFT}(\widetilde{L}_0^{(2)})}, \qquad \mathrm{FFT}(\widetilde{L}_2^{(2)}) = \overline{\mathrm{FFT}(\widetilde{L}_1^{(1)})};$$

we need sixth FFTs in the worst case, but only one single IFFT.

### 4.4.2 General refinements of the mesh

The expressions (4.33) and (4.35) can be generalized to a formula involving the nodes ${s_n}^{(j)}$ in (4.24), for $j \in \mathbb{N}$. Denoting as $[(-\Delta)^{\alpha/2}]^{(j)}$ the corresponding approximation of the fractional Laplacian, we have:

$$\begin{aligned}(-\Delta)^{\alpha/2}u\left(s_{m+\frac{1}{2}}\right) &\approx [(-\Delta)^{\alpha/2}]^{(j)}u\left(s_{m+\frac{1}{2}}\right)\\ &= \frac{c_\alpha \left|\sin\left(s_{m+\frac{1}{2}}\right)\right|^{\alpha-1}}{L^\alpha \alpha(1-\alpha)}\sum_{p=0}^{2^j-1}\left[\sum_{n=0}^{N-1} L_{p,m-n}^{(j)} g_{p,n}^{(j)}\right], \end{aligned} \tag{4.36}$$

where

$$g_{p,r}^{(j)} \equiv \sin^{1+\alpha}\left(s_{2^j r+p}^{(j)}\right) u_{ss}\left(s_{2^j r+p}^{(j)}\right) + 2\sin^{\alpha}\left(s_{2^j r+p}^{(j)}\right)\cos\left(s_{2^j r+p}^{(j)}\right) u_s\left(s_{2^j r+p}^{(j)}\right), \tag{4.37}$$

$$\begin{aligned} L_{p,r}^{(j)} &\equiv \left|\frac{\sin(h(r+1/2-(2p+1)/2^{j+1}))}{h(r+1/2-(2p+1)/2^{j+1})}\right|^{1-\alpha} K_{p,r}^{(j)}\\ &= h\left|\frac{\sin(h(r+1/2-(2p+1)/2^{j+1}))}{r+1/2-(2p+1)/2^{j+1}}\right|^{1-\alpha}\\ &\quad\left[\frac{\mathrm{sgn}(r+1/2-p/2^j)|r+1/2-p/2^j|^{2-\alpha}}{2-\alpha}\right.\\ &\quad\left.-\frac{\mathrm{sgn}(r+1/2-(p+1)/2^j)|r+1/2-(p+1)/2^j|^{2-\alpha}}{2-\alpha}\right], \end{aligned}$$

with ${s_{2^j r+p}}^{(j)} = s_{r+(2p+1)/2^{j+1}}$. Note that ${L_{2^{j+1}-1-p,r}}^{(j)} = {L_{p,-r}}^{(j)}$. We use again the fast convolution technique to compute the $2^j$ convolutions in (4.36). The procedure is identical to that of the previous examples, so we omit the details.

### 4.4.3 Numerical tests using the fast convolution and extrapolation

In order to test (4.36), we consider again $u_1(x) + iu_2(x)$ and $u_3(x)$ from Section 2.2. As we said after the definition of the nodes ${s_n}^{(j)}$ in (4.24), the nodes ${s_n}^{(j+1)}$ can be regarded as a refinement of ${s_n}^{(j)}$, and in general of $s_{n+1/2}$. Therefore, it is important to remark that, for different refinements, even if the number of ${s_n}^{(j)}$ changes with $j$, we will always use the same number of nodes $s_{n+1/2}$ in our numerical experiments.

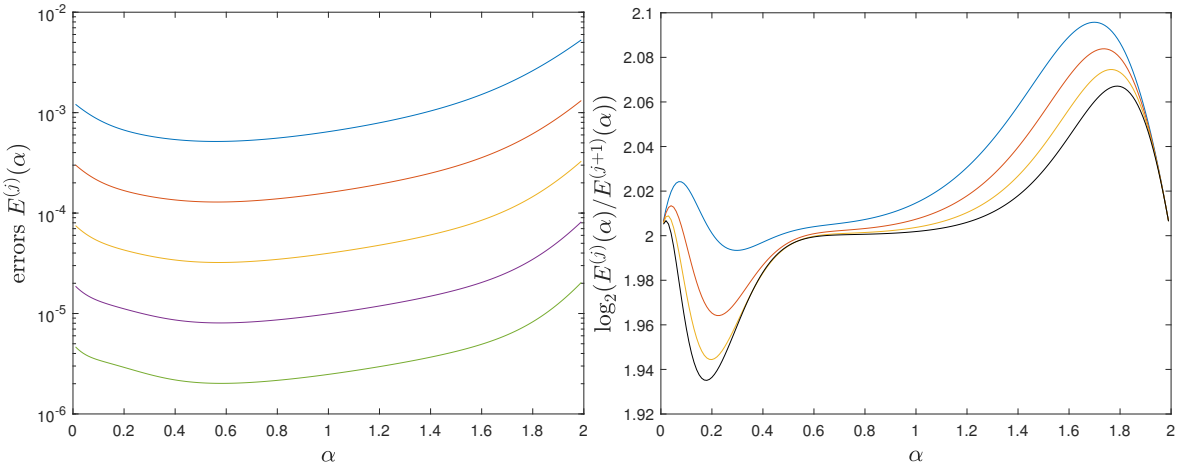


FIGURE 4.5: Left: Errors for $[-\Delta^{\alpha/2}]^{(j)}e^{i2s}$. Right: Convergence rate. We have used (4.36) and fast convolution, taking $\alpha \in \{0.01, \ldots, 0.99\} \cup \{1.01, \ldots, 1.99\}$, $N = 128$, $j = 1, \ldots, 5$.

We have computed the approximation $[(-\Delta)^{\alpha/2}]^{(j)}$ given by (4.36) of the fractional Laplacian corresponding to the $j$th refinement, for different values of $j$, compared it with the exact solution, and obtained the discrete $L^2$-norm of the corresponding error, which we denote $E^{(j)}$. In all the experiments, we have taken $N = 128$ nodes, and considered $\alpha \in \{0.01, \ldots, 0.99\} \cup \{1.01, \ldots, 1.99\}$. Observe that, unlike in the previous examples, we work now with discrete $L^2$-norm, rather than with the discrete $L^\infty$-infinity norm, because the behavior of the error for different $j$ is better appreciated, which is very important in order to adopt the correct extrapolation strategy. In this regard, recall that, given a vector of $n$ elements, $\|\mathbf{x}\|_\infty \leq \|\mathbf{x}\|_2 \leq \sqrt{n}\|\mathbf{x}\|_\infty$.

On the left-hand side of Figure 4.5, we plot against $\alpha$ the values of $E^{(j)}$ corresponding to $u_1(x) + iu_2(x) = e^{i2s}$, for $j = 1, \ldots, 5$. On the right-hand side of the same figure, we plot $\log_2(E^{(j)}/E^{(j+1)})$; and the results strongly suggest that the convergence rate is equal to 2 when $\alpha \in (0, 1) \cup (1, 2)$. We use this fact to apply Richardson extrapolation [102] to $[(-\Delta)^{\alpha/2}]^{(j)}$. More precisely, we define

$$[(-\Delta)^{\alpha/2}]^{(j,j+1)} = \frac{2^2[(-\Delta)^{\alpha/2}]^{(j+1)} - [(-\Delta)^{\alpha/2}]^{(j)}}{2^2 - 1}, \tag{4.38}$$

and denote as $E^{(j,j+1)}$ the discrete $L^2$-norm of the error of the approximation given by $[(-\Delta)^{\alpha/2}]^{(j,j+1)}$. On the left-hand side of Figure 4.6, we plot against $\alpha$ the values of $E^{(j,j+1)}$ corresponding to $u_1(x) + iu_2(x) = e^{i2s}$, for $j = 1, \ldots, 4$. On the right-hand side of the same figure, we plot $\log_2(E^{(j,j+1)}/E^{(j+1,j+2)})$. The results are not that

sharp, but they would suggest that the convergence rate behaves like $3 - |\alpha - 1|$. Therefore, we apply Richardson extrapolation a second time, by defining

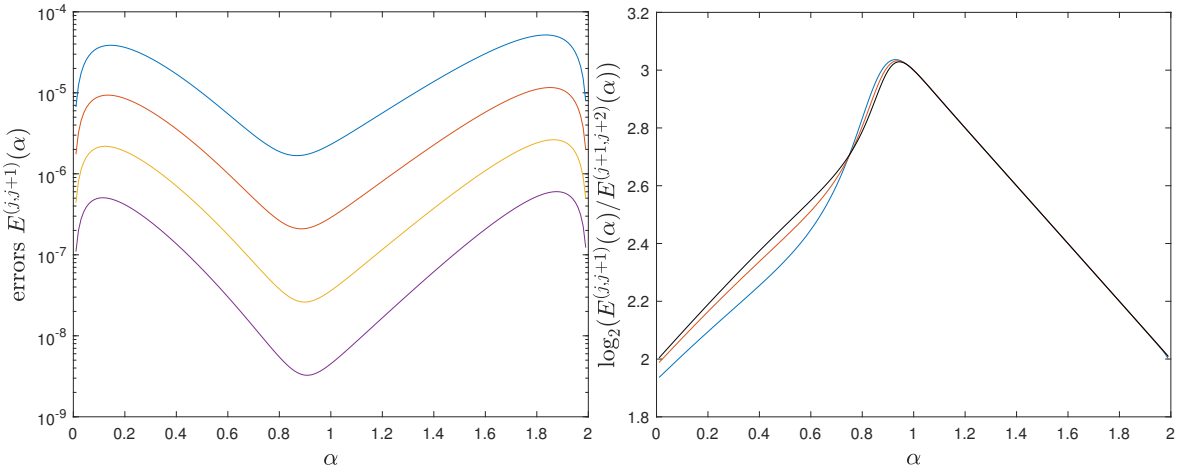


FIGURE 4.6: Left: Errors for $[-\Delta^{\alpha/2}]^{(j,j+1)}e^{i2s}$. Right: Convergence rate. We have used (4.36) and fast convolution applying Richardson extrapolation once, taking $\alpha \in \{0.01, \ldots, 0.99\} \cup \{1.01, \ldots, 1.99\}$, $N = 128$, $j = 1, \ldots, 4$.

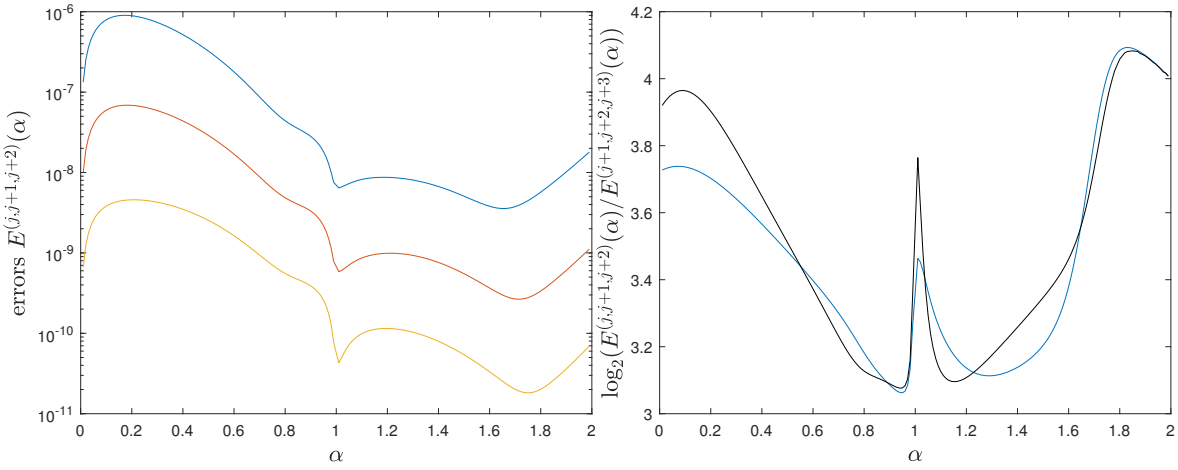


FIGURE 4.7: Left: Errors for $[-\Delta^{\alpha/2}]^{(j,j+1,j+2)}e^{i2s}$. Right: Convergence rate. We have used (4.36) and fast convolution applying Richardson extrapolation twice, taking $\alpha \in \{0.01, \ldots, 0.99\} \cup \{1.01, \ldots, 1.99\}$, $N = 128$, $j = 1, \ldots, 3$.

$$[(-\Delta)^{\alpha/2}]^{(j,j+1,j+2)} = \frac{2^{3-|\alpha-1|}[(-\Delta)^{\alpha/2}]^{(j+1,j+2)} - [(-\Delta)^{\alpha/2}]^{(j,j+1)}}{2^{3-|\alpha-1|} - 1}, \tag{4.39}$$

where $[(-\Delta)^{\alpha/2}]^{(j,j+1)}$ and $[(-\Delta)^{\alpha/2}]^{(j+1,j+2)}$ are given by (4.38). On the left-hand side of Figure 4.7, we plot against $\alpha$ the values of $E^{(j,j+1,j+2)}$ corresponding to $u_1(x) + iu_2(x) = e^{i2s}$, for $j = 1, \ldots, 3$. On the right-hand side of the same figure, we plot $\log_2(E^{(j,j+1,j+2)}/E^{(j+1,j+2,j+3)})$. The convergence rate vaguely resembles $3 + |1 - \alpha|$, but this is not concluding, so we do not apply extrapolation again.

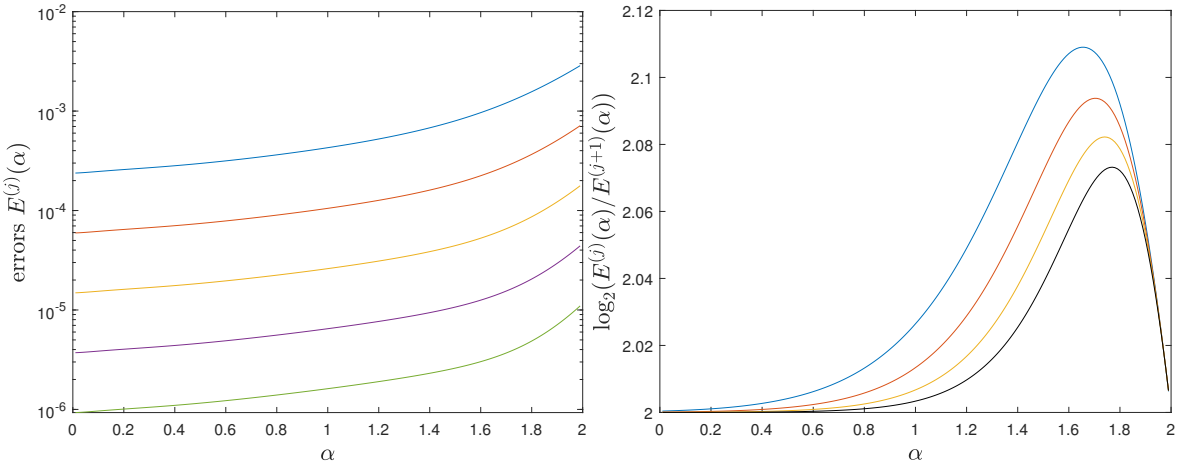


FIGURE 4.8: Left: Errors for $[-\Delta^{\alpha/2}]^{(j)}e^{-x^2}$. Right: Convergence rate. We have used (4.36) and fast convolution, taking $\alpha \in \{0.01, \ldots, 0.99\} \cup \{1.01, \ldots, 1.99\}$, $N = 128$, $j = 1, \ldots, 5$.

In what respects the fractional Laplacian of $u_3(x) = \exp(-x^2)$, on the left-hand side of Figure 4.8, we plot against $\alpha$ the values of $E^{(j)}$, for $j = 1, \ldots, 5$, and, on the right-hand side of the same figure, we plot $\log_2(E^{(j)}/E^{(j+1)})$. As in Figure 4.5, the results strongly suggest that the convergence rate is equal to 2 when $\alpha \in (0,1) \cup (1,2)$. We use this fact to apply Richardson extrapolation to $[(-\Delta)^{\alpha/2}]^{(j)}$, using (4.38) again.

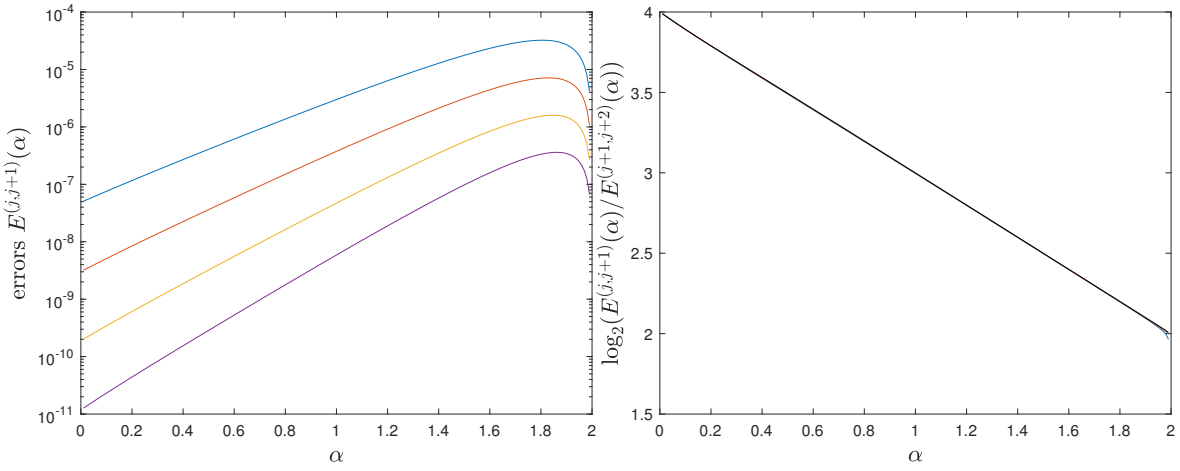


FIGURE 4.9: Left: Errors for $[-\Delta^{\alpha/2}]^{(j,j+1)}e^{-x^2}$. Right: Convergence rate. We have used (4.36) and fast convolution applying Richardson extrapolation once, taking $\alpha \in \{0.01, \ldots, 0.99\} \cup \{1.01, \ldots, 1.99\}$, $N = 128$, $j = 1, \ldots, 4$.

On the left-hand side of Figure 4.9, we plot against $\alpha$ the values of $E^{(j,j+1)}$ corresponding to $u_3(x) = \exp(-x^2)$, for $j = 1, \ldots, 4$. On the right-hand side of the same figure, we plot $\log_2(E^{(j,j+1)}/E^{(j+1,j+2)})$, and the results strongly suggest that the convergence rate is equal to $4 - \alpha$. At this point, two relevant remarks can be done: that $E^{(j)}$ seems to have a second-order convergence rate (as in Tables 4.1 and 4.2) for regular functions and, hence, (4.38) can be safely applied; and that the convergence rate of $E^{(j,j+1)}$ depends on the chosen function. For instance, in this case, (4.39) is no

longer valid, and, instead, we have to use

$$[(-\Delta)^{\alpha/2}]^{(j,j+1,j+2)} = \frac{2^{4-\alpha}[(-\Delta)^{\alpha/2}]^{(j+1,j+2)} - [(-\Delta)^{\alpha/2}]^{(j,j+1)}}{2^{4-\alpha}-1},$$

where $[(-\Delta)^{\alpha/2}]^{(j,j+1)}$ and $[(-\Delta)^{\alpha/2}]^{(j+1,j+2)}$ are again given by (4.38). We have applied Richardson extrapolation a second time, using this last formula. On the left-hand side of Figure 4.10, we plot against $\alpha$ the values of $E^{(j,j+1,j+2)}$ corresponding to $u_3(x) = \exp(-x^2)$, for $j = 1, \dots, 3$. On the right-hand side of the same figure, we plot $\log_2(E^{(j,j+1,j+2)}/E^{(j+1,j+2,j+3)})$. The convergence rate is not that sharp, but seems to be of order 4.

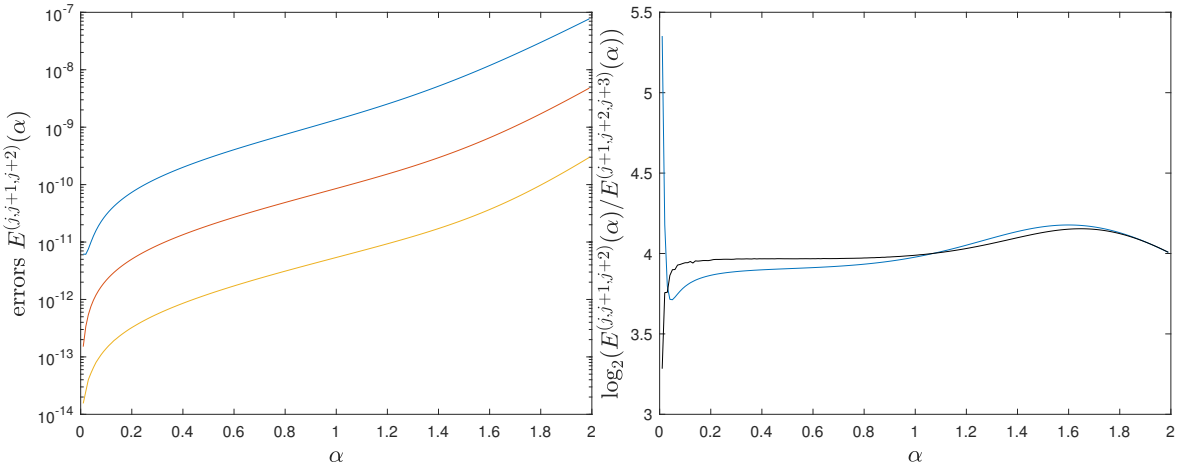


FIGURE 4.10: Left: Errors for $[-\Delta^{\alpha/2}]^{(j,j+1,j+2)}e^{-x^2}$. Right: Convergence rate. We have used (4.36) and fast convolution applying Richardson extrapolation twice, taking $\alpha \in \{0.01, \dots, 0.99\} \cup \{1.01, \dots, 1.99\}$, $N = 128$, $j = 1, \dots, 3$.

With respect to the errors, let us underline that, even without extrapolation, they can be made arbitrarily small for all $\alpha$, as we can see on the left-hand sides of Figures (4.5) and (4.8). On the other hand, unlike the methods in Sections 4.3 and 4.4, we are not changing the nodes at which we approximate the fractional Laplacian. Finally, let us mention that further research is needed, in order to prove formally that second-order convergence is always achieved, and to give an estimate of the error.

### 4.4.4 Expressing (4.36) as a single summation

In general, it is possible to give a formula equivalent to (4.36) but using only one summation symbol instead of two. In what follows, we illustrate this for the case $j = 1$, but the procedure is identical for larger $j$. Starting from (4.26), instead of dividing $[0, \pi]$ in $N$ intervals $I_n = [s_n, s_{n+1}]$, $0 \le n \le N-1$, and each of those in two subintervals $[s_n, s_{n+1/2}]$ and $[s_{n+1/2}, s_{n+1}]$, we divide directly $[0, \pi]$ in $2N$ intervals

$I_n^{(1)} = [nh/2, (n+1)h/2]$, $0 \le n \le 2N-1$:

$$(-\Delta)^{\alpha/2} u\left(s_{m+\frac{1}{2}}\right) = \frac{c_\alpha \left|\sin\left(s_{m+\frac{1}{2}}\right)\right|^{\alpha-1}}{L^\alpha \alpha(1-\alpha)} \sum_{n=0}^{2N-1} \int_{\frac{nh}{2}}^{\frac{(n+1)h}{2}} f\left(s_{m+\frac{1}{2}}, \eta\right) \left|s_{m+\frac{1}{2}} - \eta\right|^{1-\alpha} d\eta \tag{4.40}$$

Hence, if we approximate $f(s_{m+1/2}, \eta)$ in $I_n{}^{(1)}$ by its value at the middle point, i.e.,

$$f\left(s_{m+\frac{1}{2}}, s_n^{(1)}\right) \approx f\left(s_{m+\frac{1}{2}}, \eta\right), \quad \eta \in [nh/2, (n+1)h/2], \tag{4.41}$$

we get the following approximation numerically equivalent to (4.29)

$$\begin{aligned}(-\Delta)^{\alpha/2} u\left(s_{m+\frac{1}{2}}\right) &\approx \frac{c_\alpha \left|\sin\left(s_{m+\frac{1}{2}}\right)\right|^{\alpha-1}}{L^\alpha \alpha(1-\alpha)} \sum_{n=0}^{2N-1} f\left(s_{m+\frac{1}{2}}, s_n^{(1)}\right) \int_{\frac{nh}{2}}^{\frac{(n+1)h}{2}} \left|s_{m+\frac{1}{2}} - \eta\right|^{1-\alpha} d\eta \\ &= h^{2-\alpha} \frac{c_\alpha \left|\sin\left(s_{m+\frac{1}{2}}\right)\right|^{\alpha-1}}{L^\alpha \alpha(1-\alpha)} \sum_{n=0}^{2N-1} f\left(s_{m+\frac{1}{2}}, s_n^{(1)}\right) \\ &\quad \left[\frac{\operatorname{sgn}(m-n/2+1/2)|m-n/2+1/2|^{2-\alpha}}{2-\alpha} \right. \\ &\quad \left. - \frac{\operatorname{sgn}(m-n/2)|m-n/2|^{2-\alpha}}{2-\alpha}\right].\end{aligned}$$

Observe however that, unlike in (4.29), it is not completely straightforward to apply fast convolution to this last expression, so we have discarded it. Moreover, in (4.29), it is possible to use symmetries to reduce the actual number of FFTs.

## 4.5 Approximation of the fractional Laplacian using the fast convolution and the Gauss-Chebyshev quadrature

It is interesting to see what happens if we approximate the integrals in (4.40) by the midpoint rule, instead of only approximating (4.41) by a constant (and computing the integral exactly, after extracting the term containing the function $f$). This is

equivalent to applying a composite midpoint rule directly to (2.12), for $\alpha \neq 1$:

$$
\begin{aligned}
(-\Delta)^{\alpha/2} u\left(s_{m+\frac{1}{2}}\right) &\approx \frac{c_\alpha \left|\sin\left(s_{m+\frac{1}{2}}\right)\right|^{\alpha-1}}{L^\alpha \alpha(1-\alpha)} \frac{h}{2} \sum_{n=0}^{2N-1} f\left(s_{m+\frac{1}{2}}, s_n^{(1)}\right) \left|s_{m+\frac{1}{2}} - s_n^{(1)}\right|^{1-\alpha} \\
&= \frac{c_\alpha \left|\sin\left(s_{m+\frac{1}{2}}\right)\right|^{\alpha-1}}{L^\alpha \alpha(1-\alpha)} \frac{h}{2} \sum_{n=0}^{2N-1} \left|\sin\left(s_{m+\frac{1}{2}} - s_n^{(1)}\right)\right|^{1-\alpha} \\
&\quad \left( \sin^{1+\alpha}\left(s_n^{(1)}\right) u_{ss}\left(s_n^{(1)}\right) + 2\sin^{\alpha}\left(s_n^{(1)}\right) \cos\left(s_n^{(1)}\right) u_s\left(s_n^{(1)}\right) \right),
\end{aligned}
\tag{4.42}
$$

where we have expanded $f(s_{m+1/2}, {s_n}^{(1)})$ according to its definition in (4.25).

Observe that the composite midpoint rule is nothing other than the Chebyshev-Gauss quadrature in disguise [4]:

$$
\int_{-1}^{1} \frac{f(\psi)}{\sqrt{1-\psi^2}} d\psi = \sum_{j=0}^{N-1} \omega_j f(\psi_j) + R_N, \tag{4.43}
$$

where the nodes are $\psi_j = \cos(\pi(2j+1)/(2N))$, $0 \leq j \leq N-1$; the weights are constant, $\omega_j = \pi/N$; and the remainder is

$$
R_N = \frac{\pi}{(2N)! 2^{2N-1}} f^{(2N)}(\tau), \quad \tau \in (-1, 1). \tag{4.44}
$$

Indeed, under the change of variable $\psi = \cos(s)$, (4.43) becomes

$$
\int_0^{\pi} f(\cos(s)) ds = \frac{\pi}{N} \sum_{j=0}^{N-1} f\left(\frac{\pi(2j+1)}{2N}\right) + R_N.
$$

As mentioned at the beginning of Chapter 2, this quadrature formula was used in [100] to approximate $\partial_x \mathcal{D}^\alpha$, with $\mathcal{D}^\alpha$ given by (2.2), so it is natural to test it for the fractional Laplacian, too. On the other, in our case, instead of implementing (4.42), we have applied the midpoint rule to the integrals in (4.27), etc., getting:

$$
\begin{aligned}
(-\Delta)^{\alpha/2} u\left(s_{m+\frac{1}{2}}\right) \approx \frac{c_\alpha \left|\sin\left(s_{m+\frac{1}{2}}\right)\right|^{\alpha-1}}{L^\alpha \alpha(1-\alpha)} \frac{h}{2} \Bigg[ &\sum_{n=0}^{N-1} f\left(s_{m+\frac{1}{2}}, s_{2n}^{(1)}\right) \left|s_{m+\frac{1}{2}} - s_{2n}^{(1)}\right|^{1-\alpha} \\
&+ \sum_{n=0}^{N-1} f\left(s_{m+\frac{1}{2}}, s_{2n+1}^{(1)}\right) \left|s_{m+\frac{1}{2}} - s_{2n+1}^{(1)}\right|^{1-\alpha} \Bigg].
\end{aligned}
$$

Then, expanding $f(s_{m+1/2}, {s_n}^{(1)})$ and $f(s_{m+1/2}, {s_{n+1}}^{(1)})$ according to (4.25), we get the following approximation:

$$
\begin{aligned}
(-\Delta)^{\alpha/2}u\left(s_{m+\frac{1}{2}}\right) \approx \frac{c_\alpha \left|\sin\left(s_{m+\frac{1}{2}}\right)\right|^{\alpha-1}}{L^\alpha \alpha(1-\alpha)} \frac{h}{2}\Bigg[ & \sum_{n=0}^{N-1} \left|\sin\left(s_{m+\frac{1}{2}} - s_{2n}^{(1)}\right)\right|^{1-\alpha} \\
& \left( \sin^{1+\alpha}\left(s_{2n}^{(1)}\right) u_{ss}\left(s_{2n}^{(1)}\right) + 2\sin^{\alpha}\left(s_{2n}^{(1)}\right)\cos\left(s_{2n}^{(1)}\right) u_s\left(s_{2n}^{(1)}\right) \right) \\
& + \sum_{n=0}^{N-1} \left|\sin\left(s_{m+\frac{1}{2}} - s_{2n+1}^{(1)}\right)\right|^{1-\alpha} \Bigg( \sin^{1+\alpha}\left(s_{2n+1}^{(1)}\right) u_{ss}\left(s_{2n+1}^{(1)}\right) \\
& + 2\sin^{\alpha}\left(s_{2n+1}^{(1)}\right)\cos\left(s_{2n+1}^{(1)}\right) u_s\left(s_{2n+1}^{(1)}\right) \Bigg)\Bigg] \\
\approx \frac{c_\alpha \left|\sin\left(s_{m+\frac{1}{2}}\right)\right|^{\alpha-1}}{L^\alpha \alpha(1-\alpha)} & \left[ \sum_{n=0}^{N-1} P_{0,m-n}^{(1)} g_{0,n}^{(1)} + \sum_{n=0}^{N-1} P_{1,m-n}^{(1)} g_{1,n}^{(1)} \right], \qquad (4.45)
\end{aligned}
$$

i.e, we obtain a formula resembling (4.33), were ${g_0}^{(1)}$ and ${g_1}^{(1)}$, defined respectively in (4.30) and (4.31), and ${P_0}^{(0)}$ and ${P_1}^{(0)}$ are given by

$$
\begin{aligned}
P_{0,m-n}^{(1)} \equiv \frac{h}{2}\left|\sin\left(s_{m+\frac{1}{2}} - s_{2n}^{(1)}\right)\right|^{1-\alpha} &= \frac{h}{2}\left|\sin\left(h(m-n+1/4)\right)\right|^{1-\alpha} \\
\implies P_{0,r}^{(1)} &\equiv \frac{h}{2}\left|\sin(h(r+1/4))\right|^{1-\alpha}, \\
P_{1,m-n}^{(1)} \equiv \frac{h}{2}\left|\sin\left(s_{m+\frac{1}{2}} - s_{2n+1}^{(1)}\right)\right|^{1-\alpha} &= \frac{h}{2}\left|\sin\left(h(m-n-1/4)\right)\right|^{1-\alpha} \\
\implies P_{1,r}^{(1)} &\equiv \frac{h}{2}\left|\sin(h(r-1/4))\right|^{1-\alpha},
\end{aligned}
$$

where we have used that $s_{m+1/2} = h(m+1/2)$, ${s_{2n}}^{(1)} = s_{n+1/4} = h(n+1/4)$, ${s_{2n+1}}^{(1)} = s_{n+3/4} = h(n+3/4)$. Note that it is straightforward to apply the fast convolution to (4.45); since the details are identical to those of the previous examples, we omit them.

Likewise, it is possible to obtain a formula resembling (4.36); keeping the notation $[(-\Delta)^{\alpha/2}]^{(j)}$ to denote the approximation corresponding to the $j$th refinement:

$$
\begin{aligned}
(-\Delta)^{\alpha/2}u\left(s_{m+\frac{1}{2}}\right) &\approx [(-\Delta)^{\alpha/2}]^{(j)} u\left(s_{m+\frac{1}{2}}\right) \\
&= \frac{c_\alpha \left|\sin\left(s_{m+\frac{1}{2}}\right)\right|^{\alpha-1}}{L^\alpha \alpha(1-\alpha)} \sum_{p=0}^{2^j-1} \left[ \sum_{n=0}^{N-1} P_{p,m-n}^{(j)} g_{p,n}^{(j)} \right], \qquad (4.46)
\end{aligned}
$$

where ${g_p}^{(j)}$ is again defined by (4.37), and, using that ${s_{2^j n+p}}^{(j)} = s_{n+(2p+1)/2^{j+1}}$,

$$\begin{aligned} P^{(j)}_{p,m-n} &\equiv \frac{h}{2^j}\left|\sin\left(s_{m+\frac{1}{2}} - s^{(1)}_{2^j n+p}\right)\right|^{1-\alpha} \\ &= \frac{h}{2^j}\left|\sin\left(h\left(m-n+1/2-(2p+1)/2^{j+1}\right)\right)\right|^{1-\alpha} \\ &\Longrightarrow P^{(j)}_{p,r} \equiv \frac{h}{2^j}\sum_{n=0}^{N-1}\left|\sin\left(h\left(r+1/2-(2p+1)/2^{j+1}\right)\right)\right|^{1-\alpha}. \end{aligned}$$

Again, it is straightforward to apply the fast convolution to (4.46); the details can be consulted in the previous examples.

### 4.5.1 Numerical tests using the fast convolution and the Chebyshev-Gauss quadrature

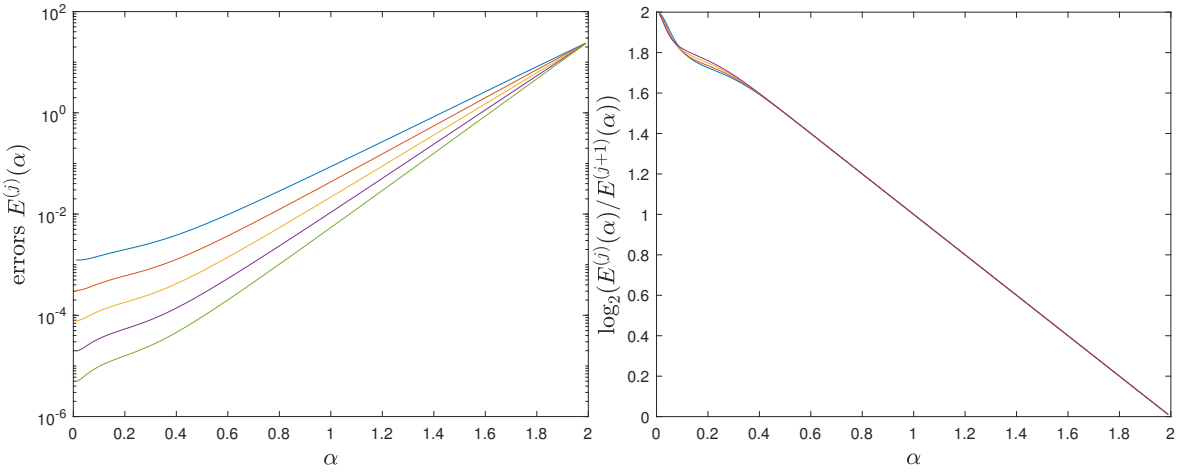


FIGURE 4.11: Left: Errors for $[-\Delta^{\alpha/2}]^{(j)}e^{i2s}$. Right: Convergence rate. We have used (4.46) and fast convolution applying Richardson, taking $\alpha \in \{0.01,\ldots,0.99\}\cup\{1.01,\ldots,1.99\}$, $N = 128$, $j = 1,\ldots,5$.

As in the previous examples, we have used in the numerical tests $u_1(x)+iu_2(x)$ and $u_3(x)$ from Section 2.2. We have computed the approximation $[(-\Delta)^{\alpha/2}]^{(j)}$ given by (4.46) of the fractional Laplacian corresponding to the $j$th refinement, for different values of $j$, compared it with the exact solution, and obtained, as in Section 4.4, the discrete $L^2$-norm of the corresponding error, which we denote again $E^{(j)}$. In all the experiments, we have taken $N = 128$ nodes, and considered $\alpha \in \{0.01,\ldots,0.99\}\cup\{1.01,\ldots,1.99\}$

On the left-hand side of Figure 4.11, we plot against $\alpha$ the values of $E^{(j)}$ corresponding to $u_1(x)+iu_2(x) = e^{i2s}$, for $j = 1,\ldots,5$. On the right-hand side of the same figure, we plot $\log_2(E^{(j)}/E^{(j+1)})$; and the results strongly suggest that the convergence rate is equal to $2-\alpha$ when $\alpha \in (0,1)\cup(1,2)$. We use this fact to apply again Richardson extrapolation to $[(-\Delta)^{\alpha/2}]^{(j)}$, defining in this case

$$[(-\Delta)^{\alpha/2}]^{(j,j+1)} = \frac{2^{2-\alpha}[(-\Delta)^{\alpha/2}]^{(j+1)} - [(-\Delta)^{\alpha/2}]^{(j)}}{2^{2-\alpha}-1}, \tag{4.47}$$

and denoting again as $E^{(j,j+1)}$ the discrete $L^2$-norm of the error of the approximation given by $[(-\Delta)^{\alpha/2}]^{(j,j+1)}$.

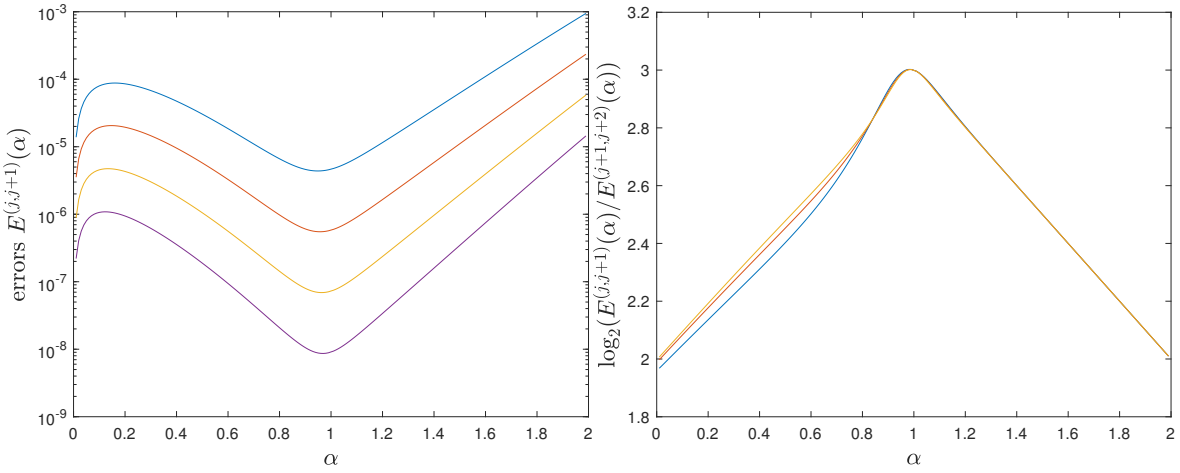


FIGURE 4.12: Left: Errors for $[-\Delta^{\alpha/2}]^{(j,j+1)}e^{i2s}$. Right: Convergence rate. We have used (4.46) and fast convolution applying Richardson extrapolation once, taking $\alpha \in \{0.01,\dots,0.99\} \cup \{1.01,\dots,1.99\}$, $N = 128$, $j = 1,\dots,4$.

On the left-hand side of Figure 4.12, we plot against $\alpha$ the values of $E^{(j,j+1)}$ corresponding to $u_1(x) + iu_2(x) = e^{i2s}$, for $j = 1,\dots,4$. On the right-hand side of the same figure, we plot $\log_2(E^{(j,j+1)}/E^{(j+1,j+2)})$, which suggest a convergence rate of $3 - |\alpha - 1|$. Therefore, we apply Richardson extrapolation a second time, by defining

$$[(-\Delta)^{\alpha/2}]^{(j,j+1,j+2)} = \frac{2^{3-|\alpha-1|}[(-\Delta)^{\alpha/2}]^{(j+1,j+2)} - [(-\Delta)^{\alpha/2}]^{(j,j+1)}}{2^{3-|\alpha-1|} - 1}.$$

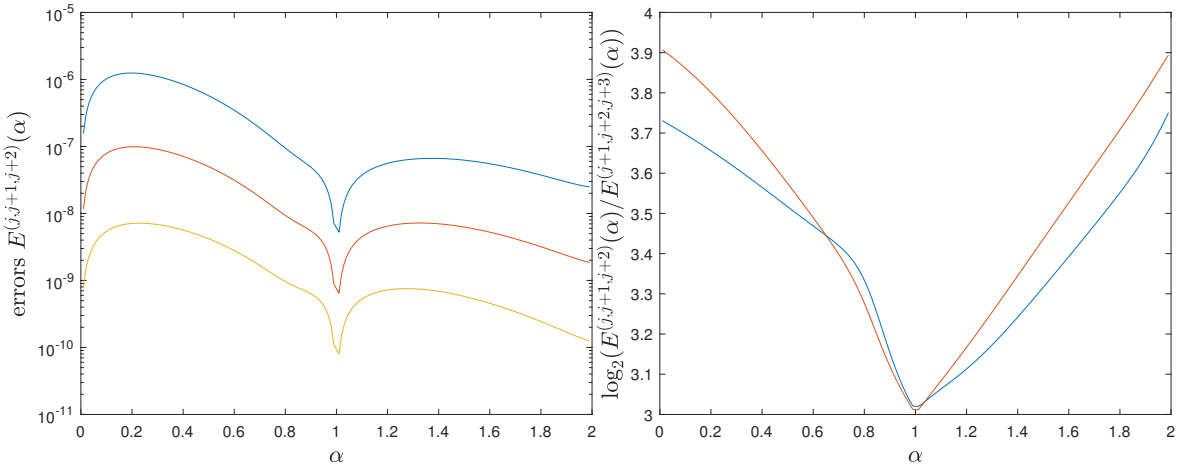


FIGURE 4.13: Left: Errors for $[-\Delta^{\alpha/2}]^{(j,j+1,j+2)}e^{i2s}$. Right: Convergence rate. We have used (4.46) and fast convolution applying Richardson extrapolation twice, taking $\alpha \in \{0.01,\dots,0.99\} \cup \{1.01,\dots,1.99\}$, $N = 128$, $j = 1,\dots,3$.

On the left-hand side of Figure 4.13, we plot against $\alpha$ the values of $E^{(j,j+1,j+2)}$ corresponding to $u_1(x) + iu_2(x) = e^{i2s}$, for $j = 1, \ldots, 3$. On the right-hand side of the same figure, we plot $\log_2(E^{(j,j+1,j+2)}/E^{(j+1,j+2,j+3)})$, which suggest a convergence rate of $3 + |1 - \alpha|$, but this is not concluding, so we do not apply extrapolation again.

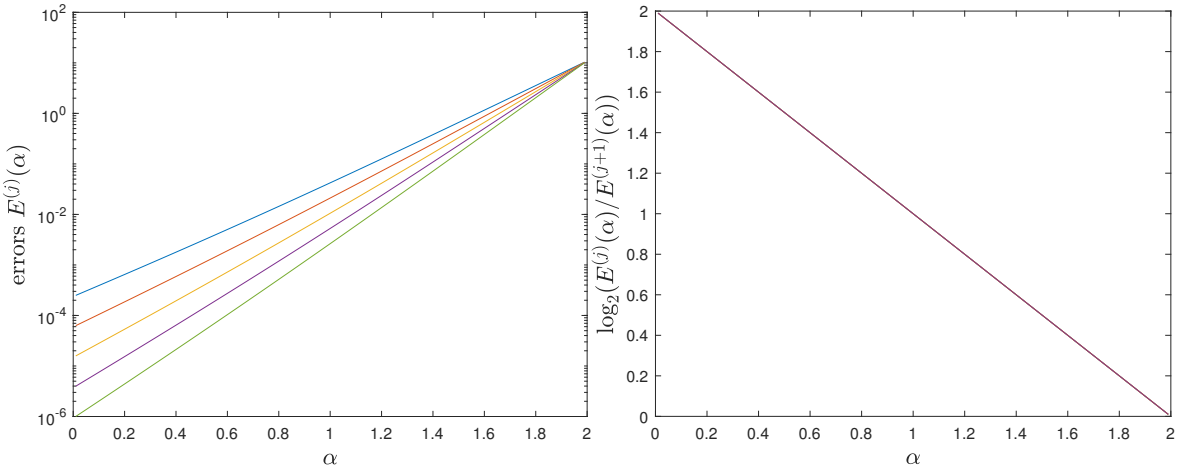


FIGURE 4.14: Left: Errors for $[-\Delta^{\alpha/2}]^{(j,j+1,j+2)}e^{-x^2}$. Right: Convergence rate. We have used (4.46) and fast convolution, taking $\alpha \in \{0.01, \ldots, 0.99\} \cup \{1.01, \ldots, 1.99\}$, $N = 128$, $j = 1, \ldots, 3$.

In what regards the fractional Laplacian of $u_3(x) = \exp(-x^2)$, on the left-hand side of Figure 4.14, we plot against $\alpha$ the values of $E^{(j)}$, for $j = 1, \ldots, 5$, and, on the right-hand side of the same figure, we plot $\log_2(E^{(j)}/E^{(j+1)})$. As in Figure 4.11, the results strongly suggest that the convergence rate is equal to $2 - \alpha$ when $\alpha \in (0, 1) \cup (1, 2)$, but the results are now even sharper; therefore, we apply Richardson extrapolation to $[(-\Delta)^{\alpha/2}]^{(j)}$, using (4.47) again.

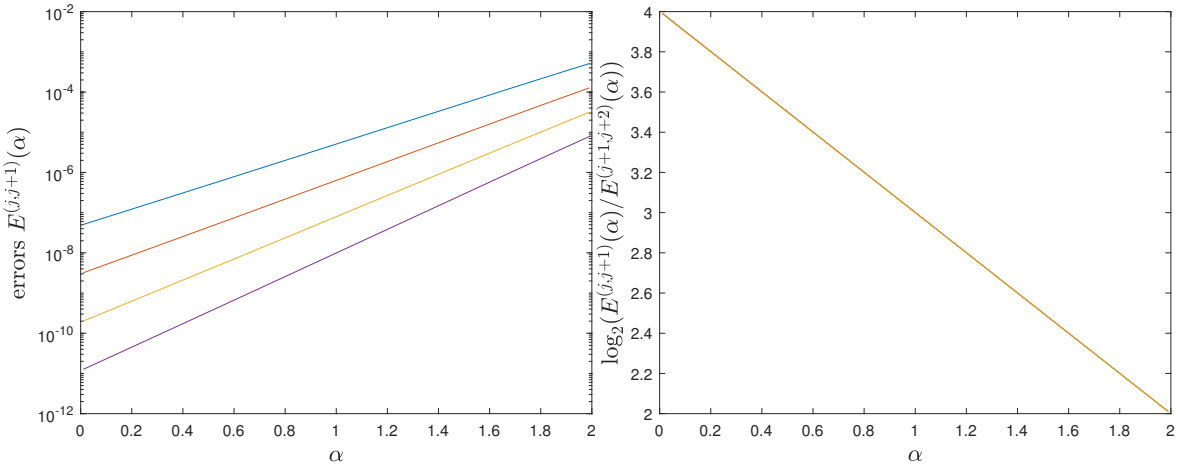


FIGURE 4.15: Left: Errors for $[-\Delta^{\alpha/2}]^{(j,j+1)}e^{-x^2}$. Right: Convergence rate. We have used (4.46) and fast convolution applying Richardson extrapolating once, taking $\alpha \in \{0.01, \ldots, 0.99\} \cup \{1.01, \ldots, 1.99\}$, $N = 128$, $j = 1, \ldots, 3$.

On the left-hand side of Figure 4.15, we plot against $\alpha$ the values of $E^{(j,j+1)}$ corresponding to $u_3(x) = \exp(-x^2)$, for $j = 1, \dots, 4$. On the right-hand side of the same figure, we plot $\log_2(E^{(j,j+1)}/E^{(j+1,j+2)})$; the results are extremely sharp and suggest that the convergence rate is equal to $4 - \alpha$. Hence, we extrapolate a second time:

$$[(-\Delta)^{\alpha/2}]^{(j,j+1,j+2)} = \frac{2^{4-\alpha}[(-\Delta)^{\alpha/2}]^{(j+1,j+2)} - [(-\Delta)^{\alpha/2}]^{(j,j+1)}}{2^{4-\alpha} - 1},$$

On the left-hand side of Figure 4.10, we plot against $\alpha$ the values of $E^{(j,j+1,j+2)}$ corresponding to $u_3(x) = \exp(-x^2)$, for $j = 1, \dots, 3$. On the right-hand side of the same figure, we plot $\log_2(E^{(j,j+1,j+2)}/E^{(j+1,j+2,j+3)})$. The convergence rate seems to be of order $6 - \alpha$, but it is not completely clear, especially for small values of $\alpha$, for which, on the other hand, the errores are smaller than $10^{-13}$.

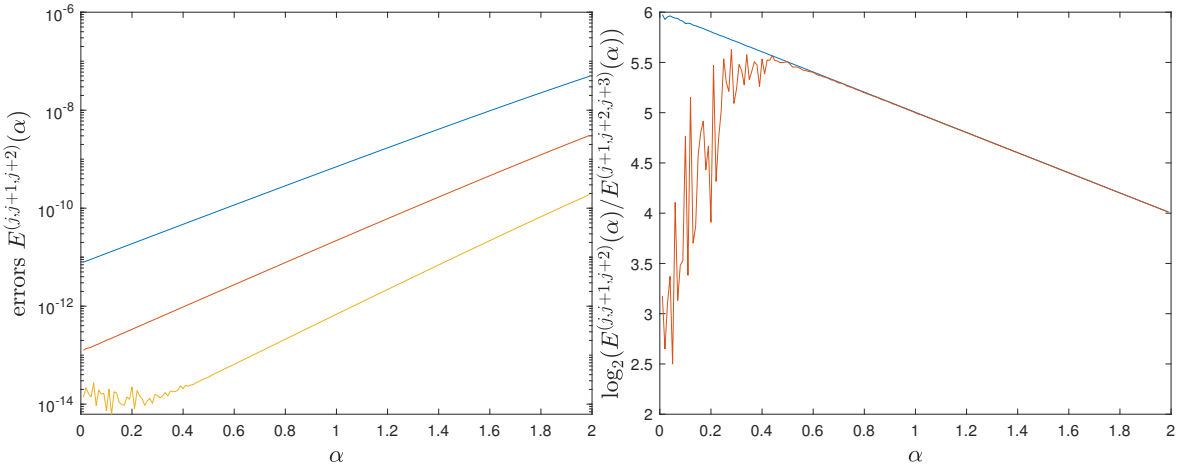


FIGURE 4.16: Left: Errors for $[-\Delta^{\alpha/2}]^{(j,j+1,j+2)}e^{-x^2}$. Right: Convergence rate. We have used (4.46) and fast convolution applying Richardson extrapolation twice, taking $\alpha \in \{0.01, \dots, 0.99\} \cup \{1.01, \dots, 1.99\}$, $N = 128$, $j = 1, \dots, 3$.

In view of the results, we can say that, without extrapolation, it is preferable to use the method in Section 4.4, because the convergence rate appears to be 2, instead of $2 - \alpha$. This is valid also when extrapolating only once, because the results of Section 4.4 are acceptable for all $\alpha$, whereas the results in this section are rather poor for certain values of $\alpha$. On the other hand, in both this section and Section 4.4, and unlike in [100], it is not clear how to extrapolate multiple times. This requires further research.

### 4.5.2 Some conclusions and future work

In this chapter, we have considered several different techniques to approximate the fractional Laplacian, and the common denominator of all of them is the use of the fast convolution; thanks to it, for a given $\alpha$, the numerical computations usually take a fraction of a second. On the other hand, we can divide the techniques in two types: those for which the points $s_{m+1/2}$ at which we approximate the fractional

Laplacian $(-\Delta)^{\alpha/2}u\,(s_{m+1/2})$ change when refining the mesh, and those for which the points do not change (i.e., we are approximating the fractional Laplacian at the same points in all the refinements). While the former techniques only use the values of $u$ and its derivatives at $s_{m+1/2}$, the latter techniques are well suited to apply Richardson's extrapolation, at least once. In this regard, we have studied two strategies to approximate $(-\Delta)^{\alpha/2}u\,(s_{m+1/2})$, which yield respectively (4.36) and (4.46). The most striking fact is that, even if the convergence rate seems to be respectively 2 and $2-\alpha$ for regular functions, the convergence order of the extrapolated formulas $[(-\Delta)^{\alpha/2}]^{(j,j+1)}$ and $[(-\Delta)^{\alpha/2}]^{(j,j+1,j+2)}$ does depend on the function used. Here, more research is needed and we postpone it for the future. However, in view of the results, it seems safe to use (4.36) even without extrapolation, because the second-order convergence enables to obtain arbitrarily small errors with a reduced computational cost.

To finish this chapter, let us enunciate a result that can serve at least as a starting point to determine the behavior of the error of (4.46) and its equivalent formula (4.42).

**Lemma 4.5.1.** *Let $w(\eta) \in \mathcal{C}^{2m+2}([0,\pi])$, such that $w(0) \neq 0$ and $w(\pi) \neq 0$. Given $a, b > -1$, we consider the following integral:*

$$I(s) = \int_0^s w(\eta)\sin^a(s-\eta)\sin^b(\eta)d\eta, \quad s \in (0,\pi).$$

*Let us approximate it numerically by the midpoint rule, i.e., given a uniform partition of $(0,s)$ of $N$ equally spaced intevals of size $h = s/N$,*

$$I(s) = h\sum_{j=0}^{N-1} \widetilde{w}\left(s,\eta_j\right)(s-\eta_j)^a\eta_j^b + E(h), \tag{4.48}$$

*where*

$$\widetilde{w}(s,\eta) = w(\eta)\left(\frac{\sin(s-\eta)}{s-\eta}\right)^a\left(\frac{\sin(\eta)}{\eta}\right)^b,$$

*and $\eta_j = jh + h/2$, for $j = 0,\ldots,N-1$. Then, there exists coefficient $C_j^{a,m}$ and $C_j^{b,m}$, such that the error $E(h)$ is*

$$E(h) = \sum_{j=1}^{2m}\left(C_j^{a,m}h^{a+j} + C_j^{b,m}h^{b+j}\right) + O(h^{\min\{a,b,1\}+2m+1}), \quad \text{as } h \to 0.$$

The proof is analogous to the one in [135] with a single power in the kernel, instead of two.

# Appendix A

# Chebyshev Polynomials and Rational Chebyshev Functions

## A.1 The Jacobi polynomials

The Jacobi polynomials $P_n^{\alpha,\beta}(x)$ of degree $n$, $n = 0, 1, 2, \ldots$, depend on the parameters $\alpha$ and $\beta$ (both $> -1$). The most immediate way to define them is through their orthogonality relation:

$$\int_{-1}^{1} P_n^{\alpha,\beta}(x) P_m^{\alpha,\beta}(x)(1-x)^{\alpha}(1+x)^{\beta} dx = 0, \quad m \neq n.$$

Theses polynomials, also known as hypergeometric polynomials, are also solutions of the Jacobi differential equation [4]:

$$(1-x^2)\frac{d^2y}{dx^2} + (\beta - \alpha - (\alpha+\beta+2)x)\frac{dy}{dx} + n(n+\alpha+\beta+1)y = 0, \tag{A.1}$$

and have some important particular cases, like the Legendre and Chebyshev polynomials [4], obtained by setting $\alpha = \beta = 0$ and $\alpha = \beta = -\frac{1}{2}$, respectively. In the following pages, we will see some properties of the Chebyshev polynomials, which constitute the basis of the rational Chebyshev functions. Remark that the properties that we mention either appear directly in [4], or can be easily derived from properties contained in [4].

### A.1.1 Definition

The Chebyshev Polynomial of the first kind $T_n(x)$ of degree $n$ has the following explicit trigonometric representation [4]:

$$T_n(x) = \cos(n\theta), \tag{A.2}$$

where $x = \cos(\theta)$, and $x \in [-1, 1]$, or equivalently, $\theta \in [0, \pi]$. The differentiation of (A.2) with respect to $x$ implies:

$$T_n'(x) = \left(\frac{d}{d\theta}\cos(n\theta)\right)\frac{d\theta}{dx} = \frac{-n\sin(n\theta)}{-\sin(\theta)} = n\frac{\sin(n\theta)}{\sin(\theta)}, \quad (x = \cos(\theta)), \tag{A.3}$$

Substituting $n$ by $n+1$, this suggests defining the following polynomial of degree $n$:

$$U_n(x) = \frac{1}{n+1} T'_{n+1}(x) = \frac{\sin((n+1)\theta)}{\sin(\theta)}, \quad (x = \cos(\theta)), \tag{A.4}$$

which is called the Chebyshev polynomial of the second kind of degree $n$ [136]. The first Chebyshev polynomials of the first and second kind are respectively [4]:

$$\begin{aligned}
T_0(x) &= 1, & U_0(x) &= 1,\\
T_1(x) &= x, & U_1(x) &= 2x,\\
T_2(x) &= 2x^2 - 1, & U_2(x) &= 4x^2 - 1,\\
T_3(x) &= 4x^3 - 3x, & U_3(x) &= 8x^3 - 4x,\\
T_4(x) &= 8x^4 - 8x^2 + 1, & U_4(x) &= 16x^4 - 12x^2 + 1,\\
T_5(x) &= 16x^5 - 20x^3 + 5x & U_5(x) &= 32x^5 - 32x^3 + 6x.
\end{aligned}$$

Let us mention some special values:

$$\begin{aligned}
T_n(-x) &= (-1)^n T_n(x), & U_n(-x) &= (-1)^n U_n(x),\\
T_n(1) &= 1, & U_n(1) &= n+1,\\
T_{2n}(0) &= (-1)^n, & U_{2n}(0) &= (-1)^n,\\
T_{2n+1}(0) &= 0, & U_{2n+1}(0) &= 0.
\end{aligned}$$

### A.1.2 Orthogonality property of $T_n(x)$ and $U_n(x)$

The Chebyshev polynomials of the first and second kind are orthogonal on $x \in [-1,1]$, with respective weights $(1-x^2)^{-1/2}$ and $(1-x^2)^{1/2}$ [4]:

$$\int_{-1}^{1} \frac{T_m(x)T_n(x)}{\sqrt{1-x^2}} dx = \begin{cases} 0, & m \neq n,\\ \dfrac{\pi}{2}, & m = n \neq 0,\\ \pi, & m = n = 0, \end{cases} \tag{A.5}$$

$$\int_{-1}^{1} U_m(x)U_n(x)\sqrt{1-x^2} dx = \begin{cases} 0, & m \neq n,\\ \dfrac{\pi}{2}, & m = n. \end{cases}$$

### A.1.3 Recurrence formulas for $T_n(x)$ and $U_n(x)$

The Chebyshev polynomials of the first and second kind have the following fundamental recurrence relation [4], for $n = 2, 3, 4, \ldots$:

$$\begin{aligned}
T_{n+1}(x) &= 2xT_n(x) - T_{n-1}(x),\\
U_{n+1}(x) &= 2xU_n(x) - U_{n-1}(x),
\end{aligned}$$

with the initial conditions $T_0(x) = U_0(x) = 1$, $T_1(x) = x$ and $U_1(x) = 2x$.

On the other hand, the recurrence formulas for $T_n'(x)$ and $U_n'(x)$ are [4]:

$$\left(1-x^2\right)\frac{d}{dx}\left(T_n(x)\right) = -nxT_n(x) + nT_{n-1}(x),$$
$$\left(1-x^2\right)\frac{d}{dx}\left(U_n(x)\right) = -nxU_n(x) + (n+1)U_{n-1}(x).$$

Moreover, there are some other recurrence relations involving both $T_n$ and $U_n$ [4]:

$$T_n(x) = U_n(x) - xU_{n-1}(x),$$
$$T_n(x) = xU_{n-1}(x) - U_{n-2}(x),$$
$$U_{n-1}(x) = \frac{1}{1-x^2}\left[xT_n(x) - T_{n+1}(x)\right].$$

### A.1.4 The Chebyshev differential equations

From the definitions (A.2) and (A.4), with $x = \cos(\theta)$, we get immediately [136]:

$$T_n'(x) = n\frac{\sin(n\theta)}{\sin(\theta)},$$
$$U_n'(x) = \frac{\sin((n+1)\theta)\cos(\theta)}{\sin^3(\theta)} - \frac{(n+1)\cos((n+1)\theta)\sin(\theta)}{\sin^3(\theta)};$$

in fact, we have mentioned the former in (A.3), as a way to motivate the definition of $U_n(x)$. Then, the second derivative on each polynomial is expressed as follows:

$$T_n''(x) = n\frac{\sin(n\theta)\cos(\theta) - n\cos(n\theta)\sin(\theta)}{\sin^3(\theta)},$$
$$U_n''(x) = \Big[3\cos^2(\theta)\sin((n+1)\theta) - n(n+2)\sin((n+1)\theta)\sin^2(\theta)$$
$$- 3(n+1)\sin(\theta)\cos(\theta)\cos((n+1)\theta)\Big]\left(\frac{1}{\sin^5(\theta)}\right).$$

Therefore, $T_n(x)$ and $U_n(x)$ satisfy respectively the following second-order ODEs:

$$(1-x^2)T_n''(x) - xT_n'(x) + n^2T_n(x) = 0,$$
$$(1-x^2)U_n''(x) - 3xU_n'(x) + n(n+2)U_n(x) = 0.$$

### A.1.5 Rodrigues' Formula

$T_n(x)$ and $U_n(x)$ can also be defined by means of Rodrigues' formula [4]:

$$T_n(x) = (-1)^n\frac{2^n n!}{(2n)!}(1-x^2)^{1/2}\frac{d^n}{dx^n}\left(1-x^2\right)^{n-1/2},$$
$$U_n(x) = (-1)^n\frac{2^n(n+1)!}{(2n+1)!}(1-x^2)^{-1/2}\frac{d^n}{dx^n}\left(1-x^2\right)^{n+1/2}.$$

### A.1.6 Chebyshev coefficients

Given a continuous function $f(x)$ defined on $[-1,1]$, there exists a family of coefficients $\{a_n\}_{n=0}^{\infty}$, such that

$$f(x) = \sum_{n=0}^{\infty} a_n P_n(x), \tag{A.6}$$

where $P_n(x)$ is either $T_n(x)$ or $U_n(x)$, and the coefficients $a_n$ are the so-called Chebyshev coefficients. In the important case, when $P_n(x) = T_n(x)$ and the series (A.6) is truncated, i.e.,

$$f(x) = \sum_{n=0}^{N-1} a_n T_n(x),$$

we can determine $a_0, \ldots, a_{N-1}$ by imposing

$$f(x_j) = \sum_{n=0}^{N-1} a_n T_n(x_j),$$

where

$$x_j = \cos\left(\frac{\pi(2j+1)}{2N}\right), \quad j = 0, \ldots, N-1,$$

are the roots of $T_N$. Therefore, we can construct an interpolating polynomial

$$p(x) = \sum_{n=0}^{N-1} a_n T_n(x),$$

such that $p(x_j) \equiv f(x_j)$, and, in general, $p(x) \approx f(x)$, when $x \in [-1,1]$. Remark that the usage of the roots $x_j$ of $T_N$ in polynomial interpolation minimizes Runge's phenomenon [137]. Moreover, if $f(x)$ is sampled at $x_j$, we obtain the corresponding coefficients $a_n$ by means of a discrete cosine transform of the values $f(x_j)$.

### A.1.7 Chebyshev coefficients for differentiation processes

Let us consider an $N$-degree polynomial $p(x)$ defined as a linear combination of the Chebyshev polynomials:

$$p(x) = \sum_{n=0}^{N} a_n T_n(x).$$

Then, its derivative is an $N-1$-degree polynomial that can also be expressed as a linear combination of those [138]:

$$p'(x) = \frac{d}{dx}\left(\sum_{n=0}^{N} a_n T_n(x)\right) = \sum_{n=0}^{N-1} a_n^{(1)} T_n(x). \tag{A.7}$$

More precisely, bearing in mind

$$2T_n(x) = \frac{T'_{n+1}(x)}{n+1} - \frac{T'_{n-1}(x)}{n-1}, \quad n = 2, 3, \ldots,$$

we get the following relationship between $a_n$ and ${a_n}^{(1)}$:

$$\mathbf{M}^{(1)} \cdot \left(a_0^{(1)}, \ldots, a_{N-1}^{(1)}\right)^T = (a_1, \ldots, a_N)^T, \tag{A.8}$$

where

$$\mathbf{M}^{(1)} = \begin{pmatrix} 1 & 0 & -\frac{1}{2} & & & & \\ \frac{1}{4} & 0 & -\frac{1}{4} & & & & \\ & \frac{1}{6} & 0 & -\frac{1}{6} & & & \\ & & \ddots & \ddots & \ddots & & \\ & & & & \frac{1}{2N-4} & 0 & -\frac{1}{2N-4} \\ & & & & & \frac{1}{2N-2} & 0 \\ & & & & & & \frac{1}{2N} \end{pmatrix}.$$

Furthermore, it is possible to express higher-order derivatives of $p(x)$ as a linear combination of the Chebyshev polynomials, by applying recursively (A.8). More precisely, if we denote the Chebyshev coefficients of the $k$th derivative by ${a_n}^{(k)}$:

$$p^{(k)}(x) = \frac{d^k}{dx^k}\left(\sum_{n=0}^{N} a_n T_n(x)\right) = \sum_{n=0}^{N-k} a_n^{(k)} T_n(x),$$

then,

$$\mathbf{M}^{(k)} \cdot \left(a_0^{(1)}, \ldots, a_{N-k}^{(1)}\right)^T = (a_k, \ldots, a_N)^T.$$

We offer $\mathbf{M}^{(2)}$, $\mathbf{M}^{(3)}$ and $\mathbf{M}^{(4)}$:

$$\mathbf{M}^{(2)} = \begin{pmatrix} \frac{1}{4} & 0 & -\frac{1}{6} & 0 & \frac{1}{24} & & & & \\ & \frac{1}{24} & 0 & -\frac{1}{16} & 0 & \frac{1}{48} & & & \\ & & \frac{1}{48} & 0 & -\frac{1}{30} & 0 & \frac{1}{80} & & \\ & & & \ddots & \ddots & \ddots & \ddots & \ddots & \\ & & & & \frac{1}{4(N-4)(N-5)} & 0 & \frac{-1}{2(N-3)(N-5)} & 0 & \frac{1}{4(N-3)(N-4)} \\ & & & & & \frac{1}{4(N-3)(N-4)} & 0 & \frac{-1}{2(N-2)(N-4)} & 0 \\ & & & & & & \frac{1}{4(N-2)(N-3)} & 0 & \frac{1}{2(N-1)(N-3)} \\ & & & & & & & \frac{1}{4(N-1)(N-2)} & 0 \\ & & & & & & & & \frac{1}{4N(N-1)} \end{pmatrix},$$

$$\mathbf{M}^{(3)} = \begin{pmatrix} \frac{1}{24} & 0 & \frac{-1}{32} & 0 & \frac{1}{80} & 0 & \frac{1}{480} & & & & \\ & \frac{1}{192} & 0 & \frac{-1}{320} & 0 & \frac{1}{196} & 0 & \frac{-1}{960} & & & \\ & & \frac{1}{480} & 0 & \frac{-1}{240} & 0 & \frac{3}{1120} & 0 & \frac{-1}{1680} & & \\ & & & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \\ & & & & \frac{1}{8(N-6)(N-7)(N-8)} & 0 & \frac{-3}{8(N-5)(N-6)(N-8)} & 0 & \frac{3}{8(N-4)(N-6)(N-7)} & 0 & \frac{-1}{8(N-4)(N-5)(N-6)} \\ & & & & & \frac{1}{8(N-5)(N-6)(N-7)} & 0 & \frac{-3}{8(N-4)(N-5)(N-7)} & 0 & \frac{3}{8(N-3)(N-5)(N-6)} & 0 \\ & & & & & & \frac{1}{8(N-4)(N-5)(N-6)} & 0 & \frac{-3}{8(N-3)(N-4)(N-6)} & 0 & \frac{3}{8(N-2)(N-4)(N-5)} \\ & & & & & & & \frac{1}{8(N-3)(N-4)(N-5)} & 0 & \frac{-3}{8(N-2)(N-3)(N-5)} & 0 \\ & & & & & & & & \frac{1}{8(N-2)(N-3)(N-4)} & 0 & \frac{-3}{8(N-1)(N-2)(N-4)} \\ & & & & & & & & & \frac{1}{8(N-1)(N-2)(N-3)} & 0 \\ & & & & & & & & & & \frac{1}{8N(N-1)(N-2)} \end{pmatrix},$$

and

$$\mathbf{M}^{(4)} = \begin{pmatrix} \frac{1}{192} & 0 & \frac{-1}{240} & 0 & \frac{3}{1440} & 0 & \frac{-1}{1680} & 0 & \frac{1}{13440} & & & & \\ & \frac{1}{1920} & 0 & \frac{-1}{960} & 0 & \frac{3}{4032} & 0 & \frac{-1}{3840} & 0 & \frac{1}{26880} & & & \\ & & \frac{1}{5760} & 0 & \frac{-1}{2520} & 0 & \frac{3}{8960} & 0 & \frac{-1}{7560} & 0 & \frac{1}{48384} & & \\ & & & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \\ & & & & \frac{1}{16(N-8)(N-9)(N-10)(N-11)} & 0 & \frac{-1}{4(N-7)(N-8)(N-9)(N-11)} & 0 & \frac{3}{8(N-6)(N-7)(N-9)(N-10)} & 0 & \frac{-1}{4(N-5)(N-7)(N-8)(N-9)} & 0 & \frac{1}{16(N-5)(N-6)(N-7)(N-8)} \\ & & & & & \frac{1}{16(N-7)(N-8)(N-9)(N-10)} & 0 & \frac{-1}{4(N-6)(N-7)(N-8)(N-10)} & 0 & \frac{3}{8(N-5)(N-6)(N-8)(N-9)} & 0 & \frac{-1}{4(N-4)(N-6)(N-7)(N-8)} & 0 \\ & & & & & & \frac{1}{16(N-6)(N-7)(N-8)(N-9)} & 0 & \frac{-1}{4(N-5)(N-6)(N-7)(N-9)} & 0 & \frac{3}{8(N-4)(N-5)(N-7)(N-8)} & 0 & \frac{-1}{4(N-3)(N-5)(N-6)(N-7)} \\ & & & & & & & \frac{1}{16(N-5)(N-6)(N-7)(N-8)} & 0 & \frac{-1}{4(N-4)(N-5)(N-6)(N-8)} & 0 & \frac{3}{8(N-3)(N-4)(N-6)(N-7)} & 0 \\ & & & & & & & & \frac{1}{16(N-4)(N-5)(N-6)(N-7)} & 0 & \frac{-1}{4(N-3)(N-4)(N-5)(N-7)} & 0 & \frac{3}{8(N-2)(N-3)(N-5)(N-6)} \\ & & & & & & & & & \frac{1}{16(N-3)(N-4)(N-5)(N-6)} & 0 & \frac{-1}{4(N-2)(N-3)(N-4)(N-6)} & 0 \\ & & & & & & & & & & \frac{1}{16(N-2)(N-3)(N-4)(N-5)} & 0 & \frac{-1}{4(N-1)(N-2)(N-3)(N-5)} \\ & & & & & & & & & & & \frac{1}{16(N-1)(N-2)(N-3)(N-4)} & 0 \\ & & & & & & & & & & & & \frac{1}{16N(N-1)(N-2)(N-3)} \end{pmatrix}.$$

### A.1.8 Differentiation through Chebyshev Matrices

For each $N \geq 1$, let the rows and columns of the Chebyshev spectral differentiation matrix $\mathbf{D}_N$ of size $(N+1) \times (N+1)$ be indexed from 0 to $N$. Then, the entries of this matrix are [139]

$$\begin{aligned} (\mathbf{D}_N)_{00} &= \frac{2N^2+1}{6}, \\ (\mathbf{D}_N)_{NN} &= -\frac{2N^2+1}{6}, \\ (\mathbf{D}_N)_{jj} &= \frac{-x_j}{2(1-x_j^2)}, \quad j = 1, \ldots, N-1, \\ (\mathbf{D}_N)_{ij} &= \frac{c_i}{c_j} \frac{(-1)^{i+j}}{(x_i - x_j)}, \quad i \neq j, \quad i, j = 0, \ldots, N, \end{aligned}$$

where

$$c_i = \begin{cases} 2, & i = 0 \text{ or } N, \\ 1, & \text{otherwise.} \end{cases}$$

and $x_j = \cos(j\pi/N)$, $j = 0, \ldots, N$ are now the points at which the extrema of $T_N(x)$ are located. The following scheme shows this pattern:

$$\mathbf{D}_N = \begin{array}{|c|c|c|} \hline \frac{2N^2+1}{6} & 2\frac{(-1)^j}{1-x_j} & \frac{1}{2}(-1)^N \\ \hline -\frac{1}{2}\frac{(-1)^i}{1-x_i} & \frac{(-1)^{i+j}}{x_i-x_j} \quad \frac{-x_j}{2(1-x_j^2)} \quad \frac{(-1)^{i+j}}{x_i-x_j} & \frac{1}{2}\frac{(-1)^{N+i}}{1+x_i} \\ \hline -\frac{1}{2}(-1)^N & -2\frac{(-1)^{N+j}}{1+x_j} & -\frac{2N^2+1}{6} \\ \hline \end{array}$$

Then, if $p(x)$ is a polynomial of degree $N$,

$$\begin{pmatrix} p'(x_0) \\ p'(x_1) \\ \vdots \\ p'(x_{N-1}) \\ p'(x_N) \end{pmatrix} = D_N \begin{pmatrix} p(x_0) \\ p(x_1) \\ \vdots \\ p(x_{N-1}) \\ p(x_N) \end{pmatrix}.$$

## A.2 Rational Chebyshev Functions

The Chebyshev polynomials are usually employed in problems involving finite intervals, but they can still be extremely useful in problems defined on e.g. $\mathbb{R}$ or $[0, \infty)$, by applying a transformation mapping $[-1, 1]$ into an infinite or semi-infinite domain (see for instance [60, 140–143]).

On the real line, the rational Chebyshev functions $TB_n(x)$, $x \in \mathbb{R}$, are defined by [2, 62]

$$TB_n(x) \equiv T_n(\psi) \equiv \cos(ns),$$

where the different domains are related through the following relationships:

$$\begin{aligned} x &= \frac{L\psi}{\sqrt{1-\psi^2}} = L\cot(s), \\ \psi &= \frac{x}{\sqrt{L^2+x^2}} \in [-1, 1], \\ s &= \operatorname{arccot}(x/L) \in [0, \pi], \end{aligned} \tag{A.9}$$

with $L > 0$ a constant chosen arbitrarily; in [2, 60, 110], some suggestions are offered to find a good choice of $L$; moreover, according to [144], it is convenient to choose $L$ to be roughly equal to the length scale of the desired solution. By applying (A.9), it is possible to simplify the programming and also understand special problems expressed by differential equations on an infinite interval [60]. The first rational Chebyshev functions are (taking $L = 1$) [144]:

$$\begin{aligned} TB_0(x) &\equiv 1, & TB_1(x) &= \frac{x}{(x^2+1)^{1/2}}, \\ TB_2(x) &= \frac{x^2-1}{x^2+1}, & TB_3(x) &= \frac{x(x^2-3)}{(x^2+1)^{3/2}}, \\ TB_4(x) &= \frac{x^4-6x^2+1}{(x^2+1)}, & TB_5(x) &= \frac{x(x^4-10y^2+5)}{(x^2+1)^{5/2}}, \end{aligned}$$

etc. As mentioned in [144], the Chebyshev functions with odd degree are not strictly speaking rational functions, because of the square root in the denominator; however, with some abuse of terminology, we will refer to all of them as the rational Chebyshev functions on $\mathbb{R}$. Let us also mention that, in [145], a Matlab function is offered to compute them.

### A.2.1 Orthogonality property

The rational Chebyshev functions are orthogonal on $\mathbb{R}$, with weight $L/(L^2+x^2)$ [2] . Therefore,

$$\int_{-\infty}^{\infty} TB_m(x)TB_n(x)\left(\frac{L}{L^2+x^2}\right)dx = \begin{cases} \pi, & m=n=0, \\ 0, & m\neq n, \\ \dfrac{\pi}{2}, & m=n>1 \end{cases}$$

### A.2.2 Conversion formulas for the derivatives under the mapping $x = L\cot(\psi)$

In order to program the rational Chebyshev functions, it is recommended to use the trigonometric representation, for which some conversion formulas for the derivatives are necessary [2]. More precisely, the first transformations of derivatives for the mapping $x = L\cot(s)$, which converts a rational Chebyshev series in $TB_n(x)$ into a Fourier cosine series in $\cos(ns)$, with $x \in \mathbb{R}$ and $s \in [0,\pi]$, are given by

$$\begin{aligned}
u_x &= -\frac{\sin^2(s)}{L}u_s, \\
u_{xx} &= \frac{\sin^3(s)}{L^2}\left[\sin(s)u_{ss} + 2\cos(s)u_s\right], \\
u_{xxx} &= -\frac{\sin^4(s)}{L^3}\left[\sin^2(s)u_{sss} + 6\cos(s)\sin(s)u_{ss} + (6-8\sin^2(s))u_s\right], \\
u_{xxxx} &= \frac{\sin^5(s)}{L^4}\Big[\sin^3(s)u_{ssss} + 12\cos(s)\sin^2(s)u_{sss} + (36\sin(s) - 44\sin^3(s))u_{ss} \\
&\quad + (24\cos(s) - 48\cos(s)\sin^2(s))u_s\Big].
\end{aligned}$$

etc. On the other hand, the first transformations of derivatives for the mapping $x = L\psi/\sqrt{1-\psi^2}$, which converts a series of $TB_n(x)$ into a Chebyshev series in $\psi$, i.e., $TB_n(x) = T_n(\psi)$, with $x \in \mathbb{R}$ and $\psi \in [-1,1]$, are given by

$$\begin{aligned}
u_x &= \frac{1-\psi^2}{L}\sqrt{1-\psi^2}u_\psi, \\
u_{xx} &= \frac{(1-\psi^2)^2}{L^2}\left[(1-\psi^2)u_{\psi\psi} - 3\psi u_\psi\right], \\
u_{xxx} &= \frac{(1-\psi^2)^{5/2}}{L^3}\left[(1-\psi^2)^2u_{\psi\psi\psi} - 9\psi(1-\psi^2)u_{\psi\psi} + (12-15(1-\psi^2))u_\psi\right], \\
u_{xxxx} &= \frac{(1-\psi^2)^3}{L^4}\Big[(1-\psi^2)^3u_{\psi\psi\psi\psi} - 18\psi(1-\psi^2)^2u_{\psi\psi\psi} \\
&\quad + (75(1-\psi^2) - 87(1-\psi^2)^2)u_{\psi\psi} + (105\psi(1-\psi^2) - 60\psi)u_\psi\Big].
\end{aligned}$$

# Bibliography


1. Kwaśnicki, M. Ten equivalent definitions of the Fractional Laplace Operator. *Fract. Calc. Appl. Anal.* **20,** 7–51 (2017).
2. Boyd, J. P. *Chebyshev and Fourier Spectral Methods* (2nd Revised Edition) (Dover Publications, 2001).
3. Frigo, M. & Johnson, S. G. The Design and Implementation of FFTW3. *Proceedings of the IEEE* **93,** 216–231 (2005).
4. Abramowitz, M. & Stegun, I. A. *Handbook of Mathematical Functions* 10th printing with corrections (Dover Publications, 1972).
5. Kolmogorov, A., Petrovsky, I. & Piscounov, N. Study of the Diffusion Equation with Growth of the Quantity of Matter and its Application to a Biology Problem. *Dynamics of Curved Fronts,* 105–130 (1988).
6. Shampine, L. F., R. C. Allen, J. & Pruess, S. *Fundamentals of Numerical Computing* (John Willey & Sons, Inc., 1997).
7. Boyd, J. P. The Orthogonal Rational Functions of Higgins and Christov and Algebraically Mapped Chebyshev Polynomials. *Journal of Approximation Theory* **61,** 98–10 (1990).
8. The MathWorks Inc. *MATLAB, Version R2017b* 2017. `https://www.mathworks.com`.
9. García-Cervera, C. J. An Efficient Real Space Method for Orbital-Free Density-Functional Theory. *Commun. Comput. Phys.* **2,** 334–357 (2007).
10. Lischke, A. *et al.* What is the fractional Laplacian? A comparative review with new results. *J. Comput. Phys.* **404,** 109009 (2020).
11. Pozrikidis, C. *The Fractional Laplacian* (CRC Press, Boca Raton, FL, 2016).
12. Hilbert, D. *Grundzüge einer allgemeinen Theorie der linearen Integralgleichungen* In German (Druck und Verlag von B. G. Teubner, Leipzig und Berlin, 1912).
13. Calderón, A. P. & Zygmund, A. On the existence of certain singular integrals. *Acta Math.* **88,** 85–139 (1952).
14. Calderón, A. P. & Zygmund, A. On Singular Integrals. *American Journal of Mathematics* **78,** 289–309 (1956).
15. Lu, S., Ding, Y. & Yan, D. *Singular Integrals and Related Topics* (World Scientific Publishing Co. Pte. Ltd., 2007).
16. Riesz, M. L'intégral de Riemann-Liouville et le probl`eme de Cauchy. *Acta Math.* **81,** 1–222 (1949).
17. Samko, S., Kilbas, A. & Marichev, O. *Fractional Integrals and Derivatives: Theory and Applications* (Switzerland: Gordon and Breach Science Publishers, 1993).
18. Chen, W. & Holm, S. Fractional Laplacian time-space models for linear and nonlinear lossy media exhibiting arbitrary frequency dependency. *The Journal of the Acoustical Society of America* **115,** 1424–1430 (2004).

19. Caffarelli, L. & Silvestre, L. An extension problem related to the fractional Laplacian. *Commun. Part. Differ. Equ.* **32,** 1245–1260 (2007).
20. Bouchaud, J.-P. & Georges, A. Anomalous diffusion in disordered media: statistical mechanisms, models and physical applications. *Phys. Rep.* **195,** 127–293 (1990).
21. Metzler, R. & Klafter, J. The random walk's guide to anomalous diffusion: a fractional dynamics approach. *Phys. Rep.* **399,** 1–77 (2000).
22. Hilfer, R. in *Applications of fractional calculus in physics* 87–130 (World Sci. Publ., River Edge, NJ, 2000).
23. Sokolov, I. M., Klafter, J. & Blumen, A. Fractional kinetics. *Phys. Today* **55,** 48–56 (2002).
24. Metzler, R. & Klafter, J. The restaurant at the end of the random walk: recent developments in the description of anomalous transport by fractional dynamics. *J. Phys. A* **37,** R161–R208 (2004).
25. Treeby, B. E. & Cox, B. T. Modeling power law absorption and dispersion for acoustic propagation using the fractional Laplacian. *The Journal of the Acoustical Society of America* **127,** 2741–2748 (2010).
26. Laskin, N. Fractional quantum mechanics and Lévy path integrals. *Phys. Lett. A* **268,** 298–305 (2000).
27. Brockmann, D. & Sokolov, I. Lévy flights in external force fields: from models to equations. *Chem. Phys.* **248,** 409–421 (2002).
28. Jespersen, S., Metzler, R. & Fogedby, H. C. Lévy flights in external force fields: Langevin and fractional Fokker-Planck equations and their solutions. *Physical Review E* **59,** 2736–2745 (1999).
29. Del Castillo-Negrete, D., Carreras, B. A. & V. E. Lynch, V. E. Front dynamics in reaction-diffusion systems with Levy flights: a fractional diffusion approach. *Physical Review Letters* **91,** 018302 (2003).
30. Baeumer, B., Kovács, M. & Meerschaert, M. M. Numerical solutions for fractional reaction-diffusion equations. *Computers & Mathematics with Applications* **55,** 2212–2226 (2008).
31. Volpert, V., Nec, Y. & Nepomnyashchy, A. A. Exact solutions in front propagation problems with superdiffusion. *Physica D: Nonlinear Phenomena* **239,** 134–144 (2010).
32. Yamamoto, M. Asymptotic expansion of solutions to the dissipative equations with fractional Laplacian. *SIAM J. Math. Anal.* **44,** 3786–3805 (2012).
33. Chmaj, A. Existence of traveling waves in the fractional bistable equation. *Archiv der Mathematik* **100,** 473–480 (2013).
34. Gui, C. & Zhao, M. Traveling wave solutions of Allen-Cahn equation with a fractional Laplacian. *Annales de l'Institut Henri Poincaré - Analyse non linéaire* **32,** 785–812 (2015).
35. Palatucci, G., Savin, O. & Valdinoci, E. Local and global minimizers for a variational energy involving a fractional norm. *Annali di Matematica Pura ed Applicata* **192,** 673–718 (2013).
36. Achleitner, F. & Kuehn, C. Analysis and numerics of traveling waves for asymmetric fractional reaction-diffusion equations. *CAIM* **6.** 25 pages (2015).
37. Achleitner, F. & Kuehn, C. Traveling waves for a bistable equation with nonlocal diffusion. *Adv. Differential Equ.* **20,** 887–936 (2015).

38. Akagi, G., Schimperna, G. & Segatti, A. Fractional Cahn-Hilliard, Allen-Cahn and porous medium equations. *J. Differ. Equations* **261,** 2935–2985 (2016).
39. Ainsworth, M. & Mao, Z. Analysis and approximation of a fractional Cahn-Hilliard equation. *SIAM J. Math. Anal.* **55,** 1689–1718 (2017).
40. Khader, M. M. & Saad, K. M. A numerical approach for solving the fractional Fisher equation using Chebyshev spectral collocation method. *Chaos Solitons Fractals* **110,** 169–177 (2018).
41. Saad, K. M., Khader, M. M., Gómez-Aguilar, J. F. & Baleanu, D. Numerical solutions of the fractional Fisher's type equations with Atangana-Baleanu fractional derivative by using spectral collocation methods. *Chaos* **29.** 023116 (2019).
42. Fisher, R. A. The wave of advance of advantageous genes. *Annals of Eugenics* **7,** 355–369 (1937).
43. Danilov, V. G., Maslov, V. P. & Vosolov, K. A. *Mathematical modelling of Heat and Mass Transfer Processes* (Kluwer, Dordrecht, 1995).
44. Polyanin, A. D. & Zaitsev, V. F. *Handbook of Nonlinear Partial Differential equation* (1st Edition) (Chapman & Hall/CRC press, 2003).
45. Starmer, C. *et al.* Vulnerability in excitable medium: Analytical and numerical studies of initiating unidirectional propagation. *Biophysical Journal* **65,** 1775–1787 (1993).
46. Keener, J. & Sneyd, J. *Mathematical physiology* (Springer, New York, 1998).
47. FitzHugh., R. Impulses and Physiological States in Theoretical Models of Nerve Membrane. *Biophysical Journal* **1,** 445–466 (1961).
48. Nagumo, J., Arimoto, S. & Yoshizawa, S. An Active Pulse Transmission Line Simulating Nerve Axon. *Proceedings of the IRE* **50,** 2061–2070 (1962).
49. Schlögl, F., Escher, C. & Berry, R. S. Fluctuations in the interface between two phases. *Physical Review A.* **27,** 2698–2704 (1983).
50. Mancinelli, R., Vergni, D. & Vulpiani, A. Superfast front propagation in reactive systems with non-Gaussian diffusion. *Europhysics Letters* **60,** 532–538 (2002).
51. Mancinelli, R., Vergni, D. & Vulpiani, A. Front propagation in reactive systems with anomalous diffusion. *Physica D: Nonlinear Phenomena* **185,** 175–195 (2003).
52. Cabré, X. & Sire, Y. Nonlinear equations for fractional Laplacians, I: Regularity, maximum principles, and Hamiltonian estimates. *Ann. I. H. Poincaré-AN* **31,** 23–53 (2014).
53. Cabré, X. & Sire, Y. Nonlinear equations for fractional Laplacians II: Existence, uniqueness, and qualitative properties of solutions. *Transactions of the American Mathematical Society* **367,** 911–941 (2015).
54. Engler, H. On the speed of spread for fractional reaction-diffusion equations. *International Journal of Differential Equations.* 16 pages (2010).
55. Cabré, X. & Roquejoffre, J.-M. Propagation de fronts dans les équations de Fisher-KPP avec diffusion fractionnaire. *Comptes Rendus Mathématique. Académie des Sciences. Paris* **347,** 1361–1366 (2009).
56. Cabré, X. & Roquejoffre, J.-M. The influence of fractional diffusion in Fisher-KPP equations. *Commun. Math. Phys.* **320,** 679–722 (2013).

57. Ilic, M., Liu, F., Turner, I. W. & Anh, V. Numerical Approximation of a Fractional-In-Space Diffusion Equation. *Fractional Calculus and Applied Analysis, An International Journal for Theory and Applicatios* **6** (2005).
58. Yang, Q., Liu, F. & Turner, I. Numerical methods for fractional partial differential equations with Riesz space fractional derivatives. *Applied Mathematical Modelling* **34,** 200–218 (2010).
59. Bueno-Orovio, A., Kay, D. & Burrage, K. Fourier spectral methods for fractional-in-space reaction-diffusion equations. *BIT Numer. Math.* **54,** 937–954 (2014).
60. Boyd, J. P. Spectral Methods Using Rational Basic Functions on an Infinite Interval. *J. Comput. Phys.* **69,** 112–142 (1987).
61. Tsynkov, S. V. Numerical solution of problems on unbounded domains. A review. *Applied Numerical Mathematics* **27,** 465–532 (1998).
62. Jovanoski, Z. & Towers, I. Application of rational Chebyshev polynomials to optical problem. *Austral. Mathematical Soc.* (2008).
63. De la Hoz, F. & Vadillo, F. A Sylvester-Based IMEX Method via Differentiation Matrices for Solving Nonlinear Parabolic Equations. *Commun. Comput. Phys.* **14,** 1001–1026 (2013).
64. Weideman, J. A. C. Computing the Hilbert transform on the real line. *Mathematics of Computation* **64,** 745–762 (1995).
65. Boyd, J. P. & Xu, Z. Comparison of three spectral methods for the Benjamin-Ono equation: Fourier pseudospectral, rational Christov functions and Gaussian radial basis functions. *Wave Motion* **48,** 702–706 (2011).
66. Huang, Y. & Oberman, A. Numerical methods for the fractional Laplacian: A finite difference-quadrature approach. *SIAM J. Numer. Anal.* **52,** 3056–3084 (2014).
67. Huang, Y. & Oberman, A. Finite difference methods for fractional Laplacians. *arXiv:1611.00164* (2016).
68. Diethelm, K., Ford, N., Freed, A. & Luchko, Y. Algorithms for the fractional calculus: a selection of numerical methods. *Comput. Methods Appl. Mech. Engrg.* **194,** 743–773 (2005).
69. Gorenflo, R. & Mainardi, F. Random walk models for space-fractional diffusion processes. *Fract. Calc. Appl. Anal.* **1,** 167–191 (1998).
70. Gorenflo, R., Fabritiis, G. & Mainardi, F. Discrete random walk models for symmetric lévy–feller diffusion processes. *Physica A: Statistical Mechanics and its Applications* **269,** 79–89 (1999).
71. Tian, X. & Du, Q. Analysis and comparison of different approximations to nonlocal diffusion and linear peridynamic equations. *SIAM J. Numer. Anal.* **51,** 3458–3482 (2013).
72. Gao, T., Duan, J., Li, X. & Song, R. Mean exit time and escape probability for dynamical systems driven by Lévy noises. *SIAM J. Sci. Comput.* **36,** A887–A906 (2014).
73. Duo, S., van Wyk, H. & Zhang, Y. A novel and accurate finite difference method for the fractional Laplacian and the fractional poisson problem. *J. Comput. Phys.* **355,** 233–252 (2018).
74. Minden, V. & Ying, L. A simple solver for the fractional Laplacian in multiple dimensions. *arXiv:1802.03770v3* (2018).

75. Fu, H., Ng, M. K. & Wang, H. A divide-and-conquer fast finite difference method for space time fractional partial differential equation. *Computers & Mathematics with Applications* **73,** 1233–1242 (2017).
76. Fu, H. & Wang, H. A preconditioned fast finite difference method for space-time fractional partial differential equations. *Fractional Calculus and Applied Analysis* **20** (2017).
77. Pang, H. & Sun, H. Multigrid method for fractional diffusion equations. *J. Comput. Phys.* **231,** 693–703 (2012).
78. Pang, H. & Sun, H. A superfast-preconditioned iterative method for steady-state space-fractional diffusion equations. *J. Comput. Phys.* **240,** 49–57 (2013).
79. Duo, S. & Zhang, Y. Accurate numerical methods for two and three dimensional integral fractional Laplacian with applications. *Comput. Methods Appl. Mech. Engrg.* **355,** 639–662 (2019).
80. Acosta, G. & Borthagaray, J. A fractional Laplace equation: Regularity of solutions and finite element approximations. *SIAM J. Numer. Anal.* **55,** 472–495 (2017).
81. Acosta, G., Bersetche, F. & Borthagaray, J. A short FE implementation for a 2d homogeneous Dirichlet problem of a fractional Laplacian. *Comput. Math. Appl.* **74,** 784–816 (2017).
82. Bonito, A., Lei, W. & Pasciak, J. Numerical approximation of the integral fractional Laplacian. *Numer. Math.* **142,** 235–278 (2019).
83. Bonito, A., Borthagaray, J., Nochetto, R., Otárola, E. & Salgado, A. Numerical methods for fractional diffusion. *Comput. Vis. Sci.* **19,** 19–46 (2018).
84. Zeng, F., Li, C., Liu, F. & Turner, I. The use of finite difference/element approaches for solving the time-fractional subdiffusion equation. *SIAM J. Sci. Comput.* **35,** A2976–A3000 (2013).
85. Mao, Z. & Karniadakis, G. A spectral method (of exponential convergence) for singular solutions of the diffusion equation with general two-sided fractional derivative. *SIAM J. Numer. Anal.* **56,** 24–29 (2018).
86. Zayernouri, M. & Karniadakis, G. Fractional spectral collocation method. *SIAM J. Sci. Comput.* **36,** A40–A62 (2014).
87. D'Elia, M. & Gunzburger, M. The fractional Laplacian operator on bounded domains as a special case of the nonlocal diffusion operator. *Comput. Math. Appl.* **66,** 1245–1260 (2013).
88. Felsinger, M., Kassmann, M. & Voigt, P. The Dirichlet problem for nonlocal operators. *Mathematische Zeitschrift* **279,** 779–809 (2015).
89. Ros-Oton, X. Nonlocal equations in bounded domains: A survey. *Publ. Mat.* **60,** 3–26 (2016).
90. Hu, Y., Li, C. & Li, H. The finite difference method for Caputo-type parabolic equation with fractional Laplacian: One-dimension case. *Chaos, Solitons & Fractals* **102,** 319–326 (2017).
91. Nochetto, R., Otárola, E. & Salgado, A. A PDE approach to fractional diffusion in general domains: a priori error analysis. *Found. Comput. Math.* **15,** 733–791 (2015).
92. Acosta, G., Borthagaray, J. P., Bruno, O. & Maas, M. Regularity theory and high order numerical methods for the (1D)-fractional Laplacian. *Math. Comp.* **87,** 1821–1857 (2018).

93. Chen, S. & Shen, J. An Efficient and Accurate Numerical Method for the Spectral Fractional Laplacian Equation. *J. Sci. Comput.* (2020).
94. Chen, L., Nochetto, R., Otárola, E. & Salgado, A. Multilevel methods for nonuniformly elliptic operators and fractional diffusion. *Math. Comput.* **85** (2016).
95. Gao, T., Duan, J., Li, X. & Song, R. Mean exit time and escape probability for dynamical systems driven by Lévy noises. *SIAM J. Sci. Comput.* **36,** A887–A906 (2014).
96. Kemppainen, M., Sjögren, P. & Torrea, J. Wave extension problem for the fractional Laplacian. *SIAM J. Sci. Comput.* **35,** 4905–4929 (2015).
97. Chen, Y., Lei, Z. & Wei, C. Extension Problems Related to the Higher Order Fractional Laplacian. *Acta Mathematica Sinica, English Series* **34,** 655–661 (2018).
98. Cayama, J., Cuesta, C. M. & de la Hoz, F. A Pseudospectral Method for the One-Dimensional Fractional Laplacian on $\mathbb{R}$. *arXiv:1908.09143* (2019).
99. Cayama, J., Cuesta, C. M. & de la Hoz, F. Numerical Approximation of the Fractional Laplacian on $\mathbb{R}$ Using Orthogonal Families. *arXiv:2001.08825* (2020).
100. De la Hoz, F. & Cuesta, C. M. A pseudo-spectral method for a non-local KdV-Burgers equation posed on $\mathbb{R}$. *J. Comput. Phys.* **311,** 45–61 (2016).
101. Kilbas, A. A, Srivastava, H. M. & Trujillo, J. J. *Theory and Applications of Fractional Differential Equations* (Elsevier, 2006).
102. Richardson, L. F. The Approximate Arithmetical Solution by Finite Differences of Physical Problems Involving Differential Equations, with an Application to the Stresses in a Masonry Dam. *Philosophical Transactions of the Royal Society A* **210,** 307–357 (1911).
103. Krasny, R. A study of singularity formation in a vortex sheet by the point-vortex approximation. *J. Fluid. Mech.* **167,** 65–93 (1986).
104. Nielsen, N. *Handbuch der Theorie der Gammafunction* In German (Druck und Verlag von B. G. Teubner, Leipzig, 1906).
105. Wolfram Research, Inc. *Mathematica, Version 11.1* 2017. `https://www.wolfram.com`.
106. Cody, J. *An Overview of Software Development for Special Functions* (Springer Verlag, Berlin, 1976).
107. Jameson, G. A simple proof of Stirling's formula for the gamma function. *The Mathematical Gazette* **99,** 68–74 (2015).
108. Morton, J. & Silverberg, L. Fourier series of half-range functions by smooth extension. *Applied Mathematical Modelling* **33,** 812–821 (2009).
109. Huybrechs, D. On the Fourier Extension of Nonperiodic Functions. *SIAM J. Numer. Anal.* **47,** 4326–4355 (2010).
110. Boyd, J. P. The Optimization of Convergence for Chebyshev Polynomial Methods in an Unbounded Domain. *J. Comput. Phys.* **45,** 43–79 (1982).
111. Kendall, D. G. A form of wave propagation associated with the equation of heat conduction. *Proc. Cambridge Philos. Soc.* **44,** 591–594 (1948).
112. Needham, D. J. A formal theory concerning the generation and propagation of travelling wave-fronts in reaction-diffusion equations. *Quart. J. Mech. Appl. Math.* **45,** 469–498 (1992).
113. Aronson, D. G. & Weinberger, H. F. Multidimensional nonlinear diffusion arising in population genetics. *Adv. Math.* **30,** 33–76 (1978).

114. Cuesta, C. M. & King, J. R. Front propagation in a heterogeneous Fisher equation: the homogeneous case is non-generic. *Quart. J. Mech. Appl. Math.* **63,** 521–571 (2010).
115. Aronson, D. G. & Weinberger, H. F. in *Partial differential equations and related topics (Program, Tulane Univ., New Orleans, La., 1974)* 5–49. Lecture Notes in Math., Vol. 446 (Springer, Berlin, 1975).
116. McKean, H. P. Application of Brownian Motion to the Equation of Kolmogorov-Petrovskii-Piskunov. *Comm. Pure Appl. Math.* **28,** 323–331 (1975).
117. McKean, H. P. A correction to: "Application of Brownian motion to the equation of Kolmogorov-Petrovskii-Piskunov". *Comm. Pure Appl. Math.* **29,** 553–554 (1976).
118. Larson, D. A. Transient bounds and time-asymptotic behavior of solutions to nonlinear equations of Fisher type. *SIAM J. Appl. Math.* **34,** 93–103 (1978).
119. Henry, D. *Geometric theory of semilinear parabolic equations* (Springer-Verlag, Berlin, 1981).
120. Sattinger, D. H. in *VII. Internationale Konferenz über Nichtlineare Schwingungen (Berlin, 1975), Band I, Teil 2* 209–213. Abh. Akad. Wissensch. DDR, Abt. Math.–Naturwissensch.–Techn., No. 4N (Akademie-Verlag, Berlin, 1977).
121. Gallay, T. Local stability of critical fronts in nonlinear parabolic partial differential equations. *Nonlinearity* **7,** 741–764 (1994).
122. Olmos, D. & Shizgal, B. D. A pseudospectral method of solution of Fisher's equation. *SIAM J. Numer. Anal. Appl. Math.* **193,** 219–242 (2006).
123. Olmos, D. & Shizgal, B. D. Pseudospectral method of solution of the Fitzhugh-Nagumo equation. *Math. Comput. Simulation* **79,** 2258–2278 (2009).
124. Needham, D. J. & Barnes, A. N. Reaction-diffusion and phase waves occurring in a class of scalar reaction-diffusion equations. *Nonlinearity* **12,** 41–58 (1999).
125. Hamel, F. & Roques, L. Fast propagation for KPP equations with slowly decaying initial conditions. *J. Differential Equations* **249,** 1726–1745 (2010).
126. Blumenson, L. A Derivation of $n$-Dimensional Spherical Coordinates. *Am. Math. Mon.* **67,** 63–66 (1960).
127. Higgins, J. R. *Completeness and Basis Properties of Sets of Special Functions* (Cambridge University Press, 1977).
128. Narayan, A. C. & Hesthaven, J. S. A generalization of the Wiener rational basis functions on infinite intervals: Part I–derivation and properties. *Mathematics and Computation* **80,** 1557–1583 (2010).
129. Christov, C. I. A Complete Orthonormal System of Functions in $L^2(-\infty, \infty)$ Space. *SIAM J. Appl. Math.* **42,** 1337–1344 (1982).
130. Wiener, N. *Extrapolation, Interpolation, and Smoothing of Stationary Time Series* (M.I.T. Press Paperback Series (Book 9), 1964).
131. Johansson, F. Computing Hypergeometric Functions Rigorously. *ACM Transactions on Mathematical Software (TOMS)* **45,** 30 (2019).
132. Advanpix LLC. *Multiprecision Computing Toolbox for MATLAB 4.7.0.13560* 2019. `http://www.advanpix.com`.
133. Ram-Mohan, L. R. *Finite Element and Boundary Element Applications in Quantum Mechanics* (Oxford University Press, 2002).

134. Press, W., Teukolsky, S., Vetterling, W. & Flannery, B. *Numerical Recipes in Fortran 77: the Art of Scientific Computing* (2nd Edition) (Cambridge University Press, New York, 1996).
135. Sidi, A. & Israeli, M. Quadrature methods for periodic singular and weakly singular Fredholm integral equations. *J. Sci. Comput.* **3,** 201–231 (1988).
136. Rivlin, T. *The Chebyshev Polynomials* (John Wiley and Sons, New York, 1974).
137. Mathews, J. H. & Fink, K. K. *Numerical Methods Using Matlab* (4rd Edition) (Pearson, 2004).
138. Fornberg, B. *A Practical Guide to Pseudospectral Methods* (Cambridge University Press, 1998).
139. Trefethen, L. N. *Spectral Methods in MATLAB* (SIAM, 2000).
140. Parand, K. & Razzaghi, M. Rational Chebyshev tau method for solving higher-order ordinary differential equations. *Int. J. Comput. Math.* **81,** 73–80 (2004).
141. Parand, K. & Razzaghi, M. Rational Chebyshev tau method for solving Volterra's population model. *Appl. Math. Comput.* **149,** 893–900 (2004).
142. Sezer, M., Gulsu, M. & Tanay, B. *Rational Chebyshev collocation method for solving higher-order linear ordinary differential equations* (Wiley Online Library, 2010).
143. Yalcinbas, S., Özsoy, N. & Sezer, M. Approximate Solution of Higher Order Linear Differential Equations by Means of a New Rational Chebyshev Collocation Method. *Mathematical and Computational Applications* **15,** 45–56 (2010).
144. Boyd, J. P. Rational Chebyshev Spectral Methods for unbounded solutions on an infinite interval using Polynomial-Growth Special Basis Functions. *Comput. Math. Appl.* **41,** 1293–1315 (2001).
145. Boyd, J. P. Weakly Nonlocal Solitary Waves and Beyond-All-Orders Asymptotics: Generalized Solitons and Hyperasymptotic Perturbation Theory. *Mathematics and Its Applications* **442** (1998).